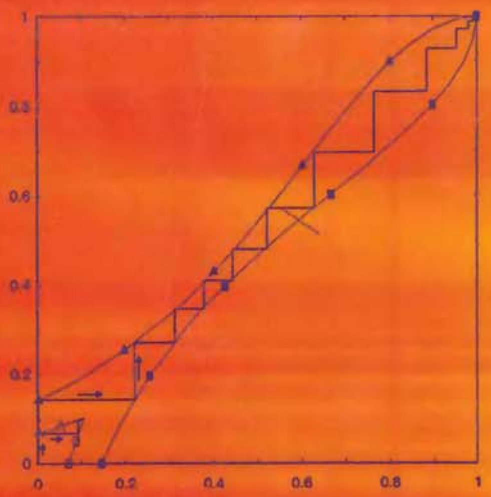
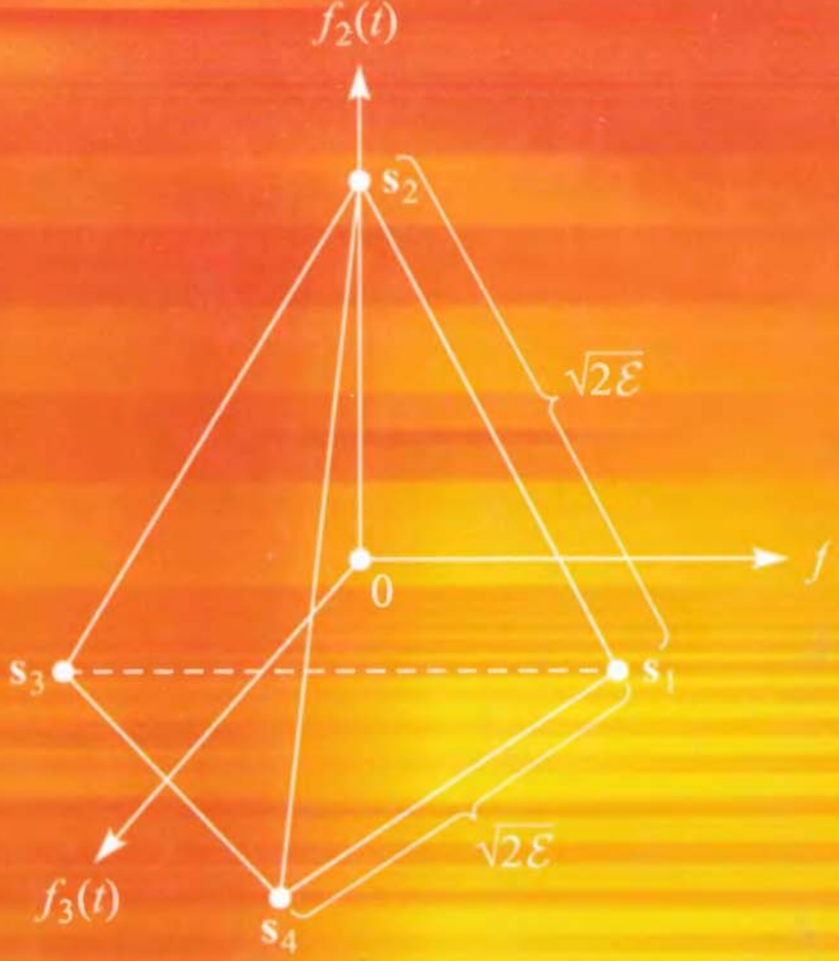
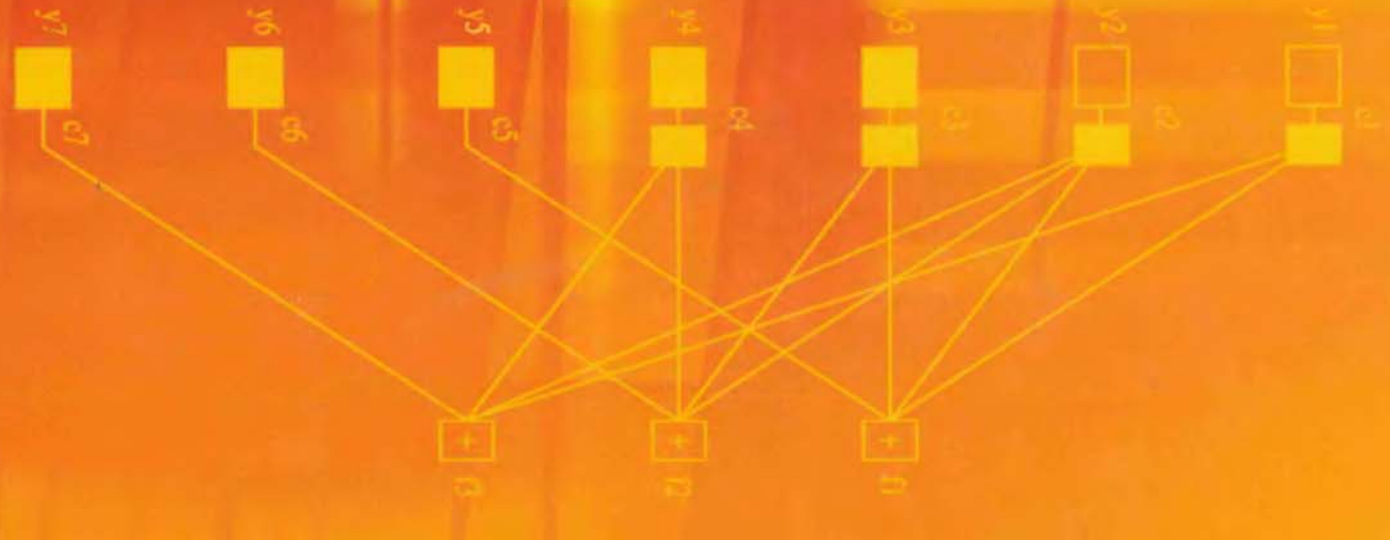
John G. Proakis
Masoud Salehi




$f_2(t)$
s_2
$\sqrt{2\mathcal{E}}$
0
f
s_3
s_1
$\sqrt{2\mathcal{E}}$
$f_3(t)$
s_4

Fifth Edition

Digital Communications


y7
y6
y5
y4
y3
y2
y1
c7
c6
c5
c4
c3
c2
c1
+
+
+
r3
r2
r1

Digital Communications

Fifth Edition

John G. Proakis
Professor Emeritus, Northeastern University
Department of Electrical and Computer Engineering,
University of California, San Diego

Masoud Salehi
Department of Electrical and Computer Engineering,
Northeastern University

McGraw-Hill
Higher Education

Boston Burr Ridge, IL Dubuque, IA New York San Francisco St. Louis
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

The McGraw·Hill Companies

McGraw-Hill
Higher Education

DIGITAL COMMUNICATIONS, FIFTH EDITION

Published by McGraw-Hill, a business unit of The McGraw-Hill Companies, Inc., 1221 Avenue of the Americas, New York, NY 10020. Copyright © 2008 by The McGraw-Hill Companies, Inc. All rights reserved. Previous editions © 2001 and 1995. No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written consent of The McGraw-Hill Companies, Inc., including, but not limited to, in any network or other electronic storage or transmission, or broadcast for distance learning.

Some ancillaries, including electronic and print components, may not be available to customers outside the United States.

This book is printed on acid-free paper.

1 2 3 4 5 6 7 8 9 0 DOC/DOC 0 9 8 7

ISBN 978-0-07-295716-7
MHID 0-07-295716-6

Global Publisher: *Raghothaman Srinivasan*
Executive Editor: *Michael Hackett*
Director of Development: *Kristine Tibbetts*
Developmental Editor: *Lorraine K. Buczek*
Executive Marketing Manager: *Michael Weitz*
Senior Project Manager: *Kay J. Brimeyer*
Lead Production Supervisor: *Sandy Ludovissy*
Associate Design Coordinator: *Brenda A. Rolwes*
Cover Designer: *Studio Montage, St. Louis, Missouri*
Compositor: *ICC Macmillan*
Typeface: *10.5/12 Times Roman*
Printer: *R. R. Donnelley Crawfordsville, IN*
(USE) Cover Image: *Chart located at top left (Figure 8.9-6): ten Brink, S. (2001). "Convergence behavior of iteratively decoded parallel concatenated codes," IEEE Transactions on Communications, vol. 49, pp.1727–1737.*

Library of Congress Cataloging-in-Publication Data

Proakis, John G.
Digital communications / John G. Proakis, Masoud Salehi.—5th ed.
p. cm.
Includes index.
ISBN 978-0-07-295716-7—ISBN 0-07-295716-6 (hbk. : alk. paper) 1. Digital communications.
I. Salehi, Masoud. II. Title.
TK5103.7.P76 2008
621.382—dc22

2007036509

www.mhhe.com

DEDICATION

To
Felia, George, and Elena
John G. Proakis

To
Fariba, Omid, Sina, and My Parents
Masoud Salehi

BRIEF CONTENTS

Preface xvi
Chapter 1 Introduction 1
Chapter 2 Deterministic and Random Signal Analysis 17
Chapter 3 Digital Modulation Schemes 95
Chapter 4 Optimum Receivers for AWGN Channels 160
Chapter 5 Carrier and Symbol Synchronization 290
Chapter 6 An Introduction to Information Theory 330
Chapter 7 Linear Block Codes 400
Chapter 8 Trellis and Graph Based Codes 491
Chapter 9 Digital Communication Through Band-Limited Channels 597
Chapter 10 Adaptive Equalization 689
Chapter 11 Multichannel and Multicarrier Systems 737
Chapter 12 Spread Spectrum Signals for Digital Communications 762
Chapter 13 Fading Channels I: Characterization and Signaling 830
Chapter 14 Fading Channels II: Capacity and Coding 899
Chapter 15 Multiple-Antenna Systems 966
Chapter 16 Multiuser Communications 1028

Appendices

Appendix A Matrices 1085
Appendix B Error Probability for Multichannel Binary Signals 1090
Appendix C Error Probabilities for Adaptive Reception of M-Phase Signals 1096
Appendix D Square Root Factorization 1107

References and Bibliography 1109
Index 1142

CONTENTS

Preface xvi

Chapter 1 Introduction **1**

1.1 Elements of a Digital Communication System 1
1.2 Communication Channels and Their Characteristics 3
1.3 Mathematical Models for Communication Channels 10
1.4 A Historical Perspective in the Development of Digital Communications 12
1.5 Overview of the Book 15
1.6 Bibliographical Notes and References 15

Chapter 2 Deterministic and Random Signal Analysis **17**

2.1 Bandpass and Lowpass Signal Representation 18
2.1–1 Bandpass and Lowpass Signals / 2.1–2 Lowpass Equivalent of Bandpass Signals / 2.1–3 Energy Considerations / 2.1–4 Lowpass Equivalent of a Bandpass System
2.2 Signal Space Representation of Waveforms 28
2.2–1 Vector Space Concepts / 2.2–2 Signal Space Concepts / 2.2–3 Orthogonal Expansions of Signals / 2.2–4 Gram-Schmidt Procedure
2.3 Some Useful Random Variables 40
2.4 Bounds on Tail Probabilities 56
2.5 Limit Theorems for Sums of Random Variables 63
2.6 Complex Random Variables 63
2.6–1 Complex Random Vectors
2.7 Random Processes 66
2.7–1 Wide-Sense Stationary Random Processes / 2.7–2 Cyclostationary Random Processes / 2.7–3 Proper and Circular Random Processes / 2.7–4 Markov Chains
2.8 Series Expansion of Random Processes 74
2.8–1 Sampling Theorem for Band-Limited Random Processes / 2.8–2 The Karhunen-Loève Expansion
2.9 Bandpass and Lowpass Random Processes 78

2.10 Bibliographical Notes and References 82
Problems 82

Chapter 3 Digital Modulation Schemes 95

3.1 Representation of Digitally Modulated Signals 95
3.2 Memoryless Modulation Methods 97
3.2–1 Pulse Amplitude Modulation (PAM) / 3.2–2 Phase Modulation / 3.2–3 Quadrature Amplitude Modulation / 3.2–4 Multidimensional Signaling
3.3 Signaling Schemes with Memory 114
3.3–1 Continuous-Phase Frequency-Shift Keying (CPFSK) / 3.3–2 Continuous-Phase Modulation (CPM)
3.4 Power Spectrum of Digitally Modulated Signals 131
3.4–1 Power Spectral Density of a Digitally Modulated Signal with Memory / 3.4–2 Power Spectral Density of Linearly Modulated Signals / 3.4–3 Power Spectral Density of Digitally Modulated Signals with Finite Memory / 3.4–4 Power Spectral Density of Modulation Schemes with a Markov Structure / 3.4–5 Power Spectral Densities of CPFSK and CPM Signals
3.5 Bibliographical Notes and References 148
Problems 148

Chapter 4 Optimum Receivers for AWGN Channels 160

4.1 Waveform and Vector Channel Models 160
4.1–1 Optimal Detection for a General Vector Channel
4.2 Waveform and Vector AWGN Channels 167
4.2–1 Optimal Detection for the Vector AWGN Channel / 4.2–2 Implementation of the Optimal Receiver for AWGN Channels / 4.2–3 A Union Bound on the Probability of Error of Maximum Likelihood Detection
4.3 Optimal Detection and Error Probability for Band-Limited Signaling 188
4.3–1 Optimal Detection and Error Probability for ASK or PAM Signaling / 4.3–2 Optimal Detection and Error Probability for PSK Signaling / 4.3–3 Optimal Detection and Error Probability for QAM Signaling / 4.3–4 Demodulation and Detection
4.4 Optimal Detection and Error Probability for Power-Limited Signaling 203
4.4–1 Optimal Detection and Error Probability for Orthogonal Signaling / 4.4–2 Optimal Detection and Error Probability for Biorthogonal Signaling / 4.4–3 Optimal Detection and Error Probability for Simplex Signaling

4.5 Optimal Detection in Presence of Uncertainty: Noncoherent Detection 210
4.5–1 Noncoherent Detection of Carrier Modulated Signals / 4.5–2 Optimal Noncoherent Detection of FSK Modulated Signals / 4.5–3 Error Probability of Orthogonal Signaling with Noncoherent Detection / 4.5–4 Probability of Error for Envelope Detection of Correlated Binary Signals / 4.5–5 Differential PSK (DPSK)
4.6 A Comparison of Digital Signaling Methods 226
4.6–1 Bandwidth and Dimensionality
4.7 Lattices and Constellations Based on Lattices 230
4.7–1 An Introduction to Lattices / 4.7–2 Signal Constellations from Lattices
4.8 Detection of Signaling Schemes with Memory 242
4.8–1 The Maximum Likelihood Sequence Detector
4.9 Optimum Receiver for CPM Signals 246
4.9–1 Optimum Demodulation and Detection of CPM / 4.9–2 Performance of CPM Signals / 4.9–3 Suboptimum Demodulation and Detection of CPM Signals
4.10 Performance Analysis for Wireline and Radio Communication Systems 259
4.10–1 Regenerative Repeaters / 4.10–2 Link Budget Analysis in Radio Communication Systems
4.11 Bibliographical Notes and References 265
Problems 266

Chapter 5 Carrier and Symbol Synchronization **290**

5.1 Signal Parameter Estimation 290
5.1–1 The Likelihood Function / 5.1–2 Carrier Recovery and Symbol Synchronization in Signal Demodulation
5.2 Carrier Phase Estimation 295
5.2–1 Maximum-Likelihood Carrier Phase Estimation / 5.2–2 The Phase-Locked Loop / 5.2–3 Effect of Additive Noise on the Phase Estimate / 5.2–4 Decision-Directed Loops / 5.2–5 Non-Decision-Directed Loops
5.3 Symbol Timing Estimation 315
5.3–1 Maximum-Likelihood Timing Estimation / 5.3–2 Non-Decision-Directed Timing Estimation
5.4 Joint Estimation of Carrier Phase and Symbol Timing 321
5.5 Performance Characteristics of ML Estimators 323
5.6 Bibliographical Notes and References 326
Problems 327

Chapter 6 An Introduction to Information Theory **330**

6.1 Mathematical Models for Information Sources 331

6.2 A Logarithmic Measure of Information 332

6.3 Lossless Coding of Information Sources 335

6.3–1 The Lossless Source Coding Theorem / 6.3–2 Lossless Coding Algorithms

6.4 Lossy Data Compression 348

6.4–1 Entropy and Mutual Information for Continuous Random Variables / 6.4–2 The Rate Distortion Function

6.5 Channel Models and Channel Capacity 354

6.5–1 Channel Models / 6.5–2 Channel Capacity

6.6 Achieving Channel Capacity with Orthogonal Signals 367

6.7 The Channel Reliability Function 369

6.8 The Channel Cutoff Rate 371

6.8–1 Bhattacharyya and Chernov Bounds / 6.8–2 Random Coding

6.9 Bibliographical Notes and References 380

Problems 381

Chapter 7 Linear Block Codes 400

7.1 Basic Definitions 401

7.1–1 The Structure of Finite Fields / 7.1–2 Vector Spaces

7.2 General Properties of Linear Block Codes 411

7.2–1 Generator and Parity Check Matrices / 7.2–2 Weight and Distance for Linear Block Codes / 7.2–3 The Weight Distribution Polynomial / 7.2–4 Error Probability of Linear Block Codes

7.3 Some Specific Linear Block Codes 420

7.3–1 Repetition Codes / 7.3–2 Hamming Codes / 7.3–3 Maximum-Length Codes / 7.3–4 Reed-Muller Codes / 7.3–5 Hadamard Codes / 7.3–6 Golay Codes

7.4 Optimum Soft Decision Decoding of Linear Block Codes 424

7.5 Hard Decision Decoding of Linear Block Codes 428

7.5–1 Error Detection and Error Correction Capability of Block Codes / 7.5–2 Block and Bit Error Probability for Hard Decision Decoding

7.6 Comparison of Performance between Hard Decision and Soft Decision Decoding 436

7.7 Bounds on Minimum Distance of Linear Block Codes 440

7.7–1 Singleton Bound / 7.7–2 Hamming Bound / 7.7–3 Plotkin Bound / 7.7–4 Elias Bound / 7.7–5 McEliece-Rodemich-Rumsey-Welch (MRRW) Bound / 7.7–6 Varshamov-Gilbert Bound

7.8 Modified Linear Block Codes 445

7.8–1 Shortening and Lengthening / 7.8–2 Puncturing and Extending / 7.8–3 Expurgation and Augmentation

7.9 Cyclic Codes 447

7.9–1 Cyclic Codes — Definition and Basic Properties / 7.9–2 Systematic Cyclic Codes / 7.9–3 Encoders for Cyclic Codes / 7.9–4 Decoding Cyclic Codes / 7.9–5 Examples of Cyclic Codes

7.10 Bose-Chaudhuri-Hocquenghem (BCH) Codes 463

7.10–1 The Structure of BCH Codes / 7.10–2 Decoding BCH Codes

7.11 Reed-Solomon Codes 471

7.12 Coding for Channels with Burst Errors 475

7.13 Combining Codes 477

7.13–1 Product Codes / 7.13–2 Concatenated Codes

7.14 Bibliographical Notes and References 482

Problems 482

Chapter 8 Trellis and Graph Based Codes 491

8.1 The Structure of Convolutional Codes 491

8.1–1 Tree, Trellis, and State Diagrams / 8.1–2 The Transfer Function of a Convolutional Code / 8.1–3 Systematic, Nonrecursive, and Recursive Convolutional Codes / 8.1–4 The Inverse of a Convolutional Encoder and Catastrophic Codes

8.2 Decoding of Convolutional Codes 510

8.2–1 Maximum-Likelihood Decoding of Convolutional Codes — The Viterbi Algorithm / 8.2–2 Probability of Error for Maximum-Likelihood Decoding of Convolutional Codes

8.3 Distance Properties of Binary Convolutional Codes 516

8.4 Punctured Convolutional Codes 516

8.4–1 Rate-Compatible Punctured Convolutional Codes

8.5 Other Decoding Algorithms for Convolutional Codes 525

8.6 Practical Considerations in the Application of Convolutional Codes 532

8.7 Nonbinary Dual-*k* Codes and Concatenated Codes 537

8.8 Maximum a Posteriori Decoding of Convolutional Codes — The BCJR Algorithm 541

8.9 Turbo Codes and Iterative Decoding 548

8.9–1 Performance Bounds for Turbo Codes / 8.9–2 Iterative Decoding for Turbo Codes / 8.9–3 EXIT Chart Study of Iterative Decoding

8.10 Factor Graphs and the Sum-Product Algorithm 558

8.10–1 Tanner Graphs / 8.10–2 Factor Graphs / 8.10–3 The Sum-Product Algorithm / 8.10–4 MAP Decoding Using the Sum-Product Algorithm

8.11 Low Density Parity Check Codes 568
8.11–1 Decoding LDPC Codes
8.12 Coding for Bandwidth-Constrained Channels — Trellis Coded Modulation 571
8.12–1 Lattices and Trellis Coded Modulation / 8.12–2 Turbo-Coded Bandwidth Efficient Modulation
8.13 Bibliographical Notes and References 589
Problems 590

Chapter 9 Digital Communication Through Band-Limited Channels 597

9.1 Characterization of Band-Limited Channels 598
9.2 Signal Design for Band-Limited Channels 602
9.2–1 Design of Band-Limited Signals for No Intersymbol Interference—The Nyquist Criterion / 9.2–2 Design of Band-Limited Signals with Controlled ISI—Partial-Response Signals / 9.2–3 Data Detection for Controlled ISI / 9.2–4 Signal Design for Channels with Distortion
9.3 Optimum Receiver for Channels with ISI and AWGN 623
9.3–1 Optimum Maximum-Likelihood Receiver / 9.3–2 A Discrete-Time Model for a Channel with ISI / 9.3–3 Maximum-Likelihood Sequence Estimation (MLSE) for the Discrete-Time White Noise Filter Model / 9.3–4 Performance of MLSE for Channels with ISI
9.4 Linear Equalization 640
9.4–1 Peak Distortion Criterion / 9.4–2 Mean-Square-Error (MSE) Criterion / 9.4–3 Performance Characteristics of the MSE Equalizer / 9.4–4 Fractionally Spaced Equalizers / 9.4–5 Baseband and Passband Linear Equalizers
9.5 Decision-Feedback Equalization 661
9.5–1 Coefficient Optimization / 9.5–2 Performance Characteristics of DFE / 9.5–3 Predictive Decision-Feedback Equalizer / 9.5–4 Equalization at the Transmitter—Tomlinson–Harashima Precoding
9.6 Reduced Complexity ML Detectors 669
9.7 Iterative Equalization and Decoding—Turbo Equalization 671
9.8 Bibliographical Notes and References 673
Problems 674

Chapter 10 Adaptive Equalization 689

10.1 Adaptive Linear Equalizer 689
10.1–1 The Zero-Forcing Algorithm / 10.1–2 The LMS Algorithm / 10.1–3 Convergence Properties of the LMS

Algorithm / 10.1–4 Excess MSE due to Noisy Gradient Estimates / 10.1–5 Accelerating the Initial Convergence Rate in the LMS Algorithm / 10.1–6 Adaptive Fractionally Spaced Equalizer—The Tap Leakage Algorithm / 10.1–7 An Adaptive Channel Estimator for ML Sequence Detection

10.2 Adaptive Decision-Feedback Equalizer 705

10.3 Adaptive Equalization of Trellis-Coded Signals 706

10.4 Recursive Least-Squares Algorithms for Adaptive Equalization 710

10.4–1 Recursive Least-Squares (Kalman) Algorithm / 10.4–2 Linear Prediction and the Lattice Filter

10.5 Self-Recovering (Blind) Equalization 721

10.5–1 Blind Equalization Based on the Maximum-Likelihood Criterion / 10.5–2 Stochastic Gradient Algorithms / 10.5–3 Blind Equalization Algorithms Based on Second- and Higher-Order Signal Statistics

10.6 Bibliographical Notes and References 731

Problems 732

Chapter 11 Multichannel and Multicarrier Systems 737

11.1 Multichannel Digital Communications in AWGN Channels 737

11.1–1 Binary Signals / 11.1–2 M-ary Orthogonal Signals

11.2 Multicarrier Communications 743

11.2–1 Single-Carrier Versus Multicarrier Modulation / 11.2–2 Capacity of a Nonideal Linear Filter Channel / 11.2–3 Orthogonal Frequency Division Multiplexing (OFDM) / 11.2–4 Modulation and Demodulation in an OFDM System / 11.2–5 An FFT Algorithm Implementation of an OFDM System / 11.2–6 Spectral Characteristics of Multicarrier Signals / 11.2–7 Bit and Power Allocation in Multicarrier Modulation / 11.2–8 Peak-to-Average Ratio in Multicarrier Modulation / 11.2–9 Channel Coding Considerations in Multicarrier Modulation

11.3 Bibliographical Notes and References 759

Problems 760

Chapter 12 Spread Spectrum Signals for Digital Communications 762

12.1 Model of Spread Spectrum Digital Communication System 763

12.2 Direct Sequence Spread Spectrum Signals 765

12.2–1 Error Rate Performance of the Decoder / 12.2–2 Some Applications of DS Spread Spectrum Signals / 12.2–3 Effect of Pulsed Interference on DS Spread

Spectrum Systems / 12.2–4 Excision of Narrowband Interference in DS Spread Spectrum Systems / 12.2–5 Generation of PN Sequences

12.3 Frequency-Hopped Spread Spectrum Signals 802

12.3–1 Performance of FH Spread Spectrum Signals in an AWGN Channel / 12.3–2 Performance of FH Spread Spectrum Signals in Partial-Band Interference / 12.3–3 A CDMA System Based on FH Spread Spectrum Signals

12.4 Other Types of Spread Spectrum Signals 814

12.5 Synchronization of Spread Spectrum Systems 815

12.6 Bibliographical Notes and References 823

Problems 823

Chapter 13 Fading Channels I: Characterization and Signaling 830

13.1 Characterization of Fading Multipath Channels 831

13.1–1 Channel Correlation Functions and Power Spectra / 13.1–2 Statistical Models for Fading Channels

13.2 The Effect of Signal Characteristics on the Choice of a Channel Model 844

13.3 Frequency-Nonselective, Slowly Fading Channel 846

13.4 Diversity Techniques for Fading Multipath Channels 850

13.4–1 Binary Signals / 13.4–2 Multiphase Signals / 13.4–3 M-ary Orthogonal Signals

13.5 Signaling over a Frequency-Selective, Slowly Fading Channel: The RAKE Demodulator 869

13.5–1 A Tapped-Delay-Line Channel Model / 13.5–2 The RAKE Demodulator / 13.5–3 Performance of RAKE Demodulator / 13.5–4 Receiver Structures for Channels with Intersymbol Interference

13.6 Multicarrier Modulation (OFDM) 884

13.6–1 Performance Degradation of an OFDM System due to Doppler Spreading / 13.6–2 Suppression of ICI in OFDM Systems

13.7 Bibliographical Notes and References 890

Problems 891

Chapter 14 Fading Channels II: Capacity and Coding 899

14.1 Capacity of Fading Channels 900

14.1–1 Capacity of Finite-State Channels

14.2 Ergodic and Outage Capacity 905

14.2–1 The Ergodic Capacity of the Rayleigh Fading Channel / 14.2–2 The Outage Capacity of Rayleigh Fading Channels

14.3 Coding for Fading Channels 918

14.4 Performance of Coded Systems In Fading Channels 919
14.4–1 Coding for Fully Interleaved Channel Model
14.5 Trellis-Coded Modulation for Fading Channels 929
14.5–1 TCM Systems for Fading Channels / 14.5–2 Multiple Trellis-Coded Modulation (MTCM)
14.6 Bit-Interleaved Coded Modulation 936
14.7 Coding in the Frequency Domain 942
14.7–1 Probability of Error for Soft Decision Decoding of Linear Binary Block Codes / 14.7–2 Probability of Error for Hard-Decision Decoding of Linear Block Codes / 14.7–3 Upper Bounds on the Performance of Convolutional Codes for a Rayleigh Fading Channel / 14.7–4 Use of Constant-Weight Codes and Concatenated Codes for a Fading Channel
14.8 The Channel Cutoff Rate for Fading Channels 957
14.8–1 Channel Cutoff Rate for Fully Interleaved Fading Channels with CSI at Receiver
14.9 Bibliographical Notes and References 960
Problems 961

Chapter 15 Multiple-Antenna Systems 966

15.1 Channel Models for Multiple-Antenna Systems 966
15.1–1 Signal Transmission Through a Slow Fading Frequency-Nonselective MIMO Channel / 15.1–2 Detection of Data Symbols in a MIMO System / 15.1–3 Signal Transmission Through a Slow Fading Frequency-Selective MIMO Channel
15.2 Capacity of MIMO Channels 981
15.2–1 Mathematical Preliminaries / 15.2–2 Capacity of a Frequency-Nonselective Deterministic MIMO Channel / 15.2–3 Capacity of a Frequency-Nonselective Ergodic Random MIMO Channel / 15.2–4 Outage Capacity / 15.2–5 Capacity of MIMO Channel When the Channel Is Known at the Transmitter
15.3 Spread Spectrum Signals and Multicode Transmission 992
15.3–1 Orthogonal Spreading Sequences / 15.3–2 Multiplexing Gain Versus Diversity Gain / 15.3–3 Multicode MIMO Systems
15.4 Coding for MIMO Channels 1001
15.4–1 Performance of Temporally Coded SISO Systems in Rayleigh Fading Channels / 15.4–2 Bit-Interleaved Temporal Coding for MIMO Channels / 15.4–3 Space-Time Block Codes for MIMO Channels / 15.4–4 Pairwise Error Probability for a Space-Time Code / 15.4–5 Space-Time Trellis Codes for MIMO Channels / 15.4–6 Concatenated Space-Time Codes and Turbo Codes

15.5 Bibliographical Notes and References 1021
Problems 1021

Chapter 16 Multiuser Communications 1028

16.1 Introduction to Multiple Access Techniques 1028
16.2 Capacity of Multiple Access Methods 1031
16.3 Multiuser Detection in CDMA Systems 1036
16.3–1 CDMA Signal and Channel Models / 16.3–2 The Optimum Multiuser Receiver / 16.3–3 Suboptimum Detectors / 16.3–4 Successive Interference Cancellation / 16.3–5 Other Types of Multiuser Detectors / 16.3–6 Performance Characteristics of Detectors
16.4 Multiuser MIMO Systems for Broadcast Channels 1053
16.4–1 Linear Precoding of the Transmitted Signals / 16.4–2 Nonlinear Precoding of the Transmitted Signals—The QR Decomposition / 16.4–3 Nonlinear Vector Precoding / 16.4–4 Lattice Reduction Technique for Precoding
16.5 Random Access Methods 1068
16.5–1 ALOHA Systems and Protocols / 16.5–2 Carrier Sense Systems and Protocols
16.6 Bibliographical Notes and References 1077
Problems 1078

Appendix A Matrices **1085**
A.1 Eigenvalues and Eigenvectors of a Matrix 1086
A.2 Singular-Value Decomposition 1087
A.3 Matrix Norm and Condition Number 1088
A.4 The Moore–Penrose Pseudoinverse 1088

Appendix B Error Probability for Multichannel Binary Signals **1090**

Appendix C Error Probabilities for Adaptive Reception of M-Phase Signals **1096**
C.1 Mathematical Model for an M-Phase Signaling Communication System 1096
C.2 Characteristic Function and Probability Density Function of the Phase θ 1098
C.3 Error Probabilities for Slowly Fading Rayleigh Channels 1100
C.4 Error Probabilities for Time-Invariant and Ricean Fading Channels 1104

Appendix D Square Root Factorization **1107**

References and Bibliography 1109

Index 1142

PREFACE

It is a pleasure to welcome Professor Masoud Salehi as a coauthor to the fifth edition of *Digital Communications.* This new edition has undergone a major revision and reorganization of topics, especially in the area of channel coding and decoding. A new chapter on multiple-antenna systems has been added as well.

The book is designed to serve as a text for a first-year graduate-level course for students in electrical engineering. It is also designed to serve as a text for self-study and as a reference book for the practicing engineer involved in the design and analysis of digital communications systems. As to background, we presume that the reader has a thorough understanding of basic calculus and elementary linear systems theory and prior knowledge of probability and stochastic processes.

Chapter 1 is an introduction to the subject, including a historical perspective and a description of channel characteristics and channel models.

Chapter 2 contains a review of deterministic and random signal analysis, including bandpass and lowpass signal representations, bounds on the tail probabilities of random variables, limit theorems for sums of random variables, and random processes.

Chapter 3 treats digital modulation techniques and the power spectrum of digitally modulated signals.

Chapter 4 is focused on optimum receivers for additive white Gaussian noise (AWGN) channels and their error rate performance. Also included in this chapter is an introduction to lattices and signal constellations based on lattices, as well as link budget analyses for wireline and radio communication systems.

Chapter 5 is devoted to carrier phase estimation and time synchronization methods based on the maximum-likelihood criterion. Both decision-directed and non-decision-directed methods are described.

Chapter 6 provides an introduction to topics in information theory, including lossless source coding, lossy data compression, channel capacity for different channel models, and the channel reliability function.

Chapter 7 treats linear block codes and their properties. Included is a treatment of cyclic codes, BCH codes, Reed-Solomon codes, and concatenated codes. Both soft decision and hard decision decoding methods are described, and their performance in AWGN channels is evaluated.

Chapter 8 provides a treatment of trellis codes and graph-based codes, including convolutional codes, turbo codes, low density parity check (LDPC) codes, trellis codes for band-limited channels, and codes based on lattices. Decoding algorithms are also treated, including the Viterbi algorithm and its performance on AWGN channels, the BCJR algorithm for iterative decoding of turbo codes, and the sum-product algorithm.

Chapter 9 is focused on digital communication through band-limited channels. Topics treated in this chapter include the characterization and signal design for band-limited channels, the optimum receiver for channels with intersymbol interference and AWGN, and suboptimum equalization methods, namely, linear equalization, decision-feedback equalization, and turbo equalization.

Chapter 10 treats adaptive channel equalization. The LMS and recursive least-squares algorithms are described together with their performance characteristics. This chapter also includes a treatment of blind equalization algorithms.

Chapter 11 provides a treatment of multichannel and multicarrier modulation. Topics treated include the error rate performance of multichannel binary signal and M-ary orthogonal signals in AWGN channels; the capacity of a nonideal linear filter channel with AWGN; OFDM modulation and demodulation; bit and power allocation in an OFDM system; and methods to reduce the peak-to-average power ratio in OFDM.

Chapter 12 is focused on spread spectrum signals and systems, with emphasis on direct sequence and frequency-hopped spread spectrum systems and their performance. The benefits of coding in the design of spread spectrum signals is emphasized throughout this chapter.

Chapter 13 treats communication through fading channels, including the characterization of fading channels and the key important parameters of multipath spread and Doppler spread. Several channel fading statistical models are introduced, with emphasis placed on Rayleigh fading, Ricean fading, and Nakagami fading. An analysis of the performance degradation caused by Doppler spread in an OFDM system is presented, and a method for reducing this performance degradation is described.

Chapter 14 is focused on capacity and code design for fading channels. After introducing ergodic and outage capacities, coding for fading channels is studied. Bandwidth-efficient coding and bit-interleaved coded modulation are treated, and the performance of coded systems in Rayleigh and Ricean fading is derived.

Chapter 15 provides a treatment of multiple-antenna systems, generally called multiple-input, multiple-output (MIMO) systems, which are designed to yield spatial signal diversity and spatial multiplexing. Topics treated in this chapter include detection algorithms for MIMO channels, the capacity of MIMO channels with AWGN without and with signal fading, and space-time coding.

Chapter 16 treats multiuser communications, including the topics of the capacity of multiple-access methods, multiuser detection methods for the uplink in CDMA systems, interference mitigation in multiuser broadcast channels, and random access methods such as ALOHA and carrier-sense multiple access (CSMA).

With 16 chapters and a variety of topics, the instructor has the flexibility to design either a one- or two-semester course. Chapters 3, 4, and 5 provide a basic treatment of digital modulation/demodulation and detection methods. Channel coding and decoding treated in Chapters 7, 8, and 9 can be included along with modulation/demodulation in a one-semester course. Alternatively, Chapters 9 through 12 can be covered in place of channel coding and decoding. A second semester course can cover the topics of communication through fading channels, multiple-antenna systems, and multiuser communications.

The authors and McGraw-Hill would like to thank the following reviewers for their suggestions on selected chapters of the fifth edition manuscript:

Paul Salama, *Indiana University/Purdue University, Indianapolis;* Dimitrios Hatzinakos, *University of Toronto,* and Ender Ayanoglu, *University of California, Irvine.*

Finally, the first author wishes to thank Gloria Doukakis for her assistance in typing parts of the manuscript. We also thank Patrick Amihood for preparing several graphs in Chapters 15 and 16 and Apostolos Rizos and Kostas Stamatiou for preparing parts of the Solutions Manual.

1

Introduction

In this book, we present the basic principles that underlie the analysis and design of digital communication systems. The subject of digital communications involves the transmission of information in digital form from a source that generates the information to one or more destinations. Of particular importance in the analysis and design of communication systems are the characteristics of the physical channels through which the information is transmitted. The characteristics of the channel generally affect the design of the basic building blocks of the communication system. Below, we describe the elements of a communication system and their functions.

1.1

ELEMENTS OF A DIGITAL COMMUNICATION SYSTEM

Figure 1.1–1 illustrates the functional diagram and the basic elements of a digital communication system. The source output may be either an analog signal, such as an audio or video signal, or a digital signal, such as the output of a computer, that is discrete in time and has a finite number of output characters. In a digital communication system, the messages produced by the source are converted into a sequence of binary digits. Ideally, we should like to represent the source output (message) by as few binary digits as possible. In other words, we seek an efficient representation of the source output that results in little or no redundancy. The process of efficiently converting the output of either an analog or digital source into a sequence of binary digits is called *source encoding* or *data compression*.

The sequence of binary digits from the source encoder, which we call the *information sequence*, is passed to the *channel encoder*. The purpose of the channel encoder is to introduce, in a controlled manner, some redundancy in the binary information sequence that can be used at the receiver to overcome the effects of noise and interference encountered in the transmission of the signal through the channel. Thus, the added redundancy serves to increase the reliability of the received data and improves

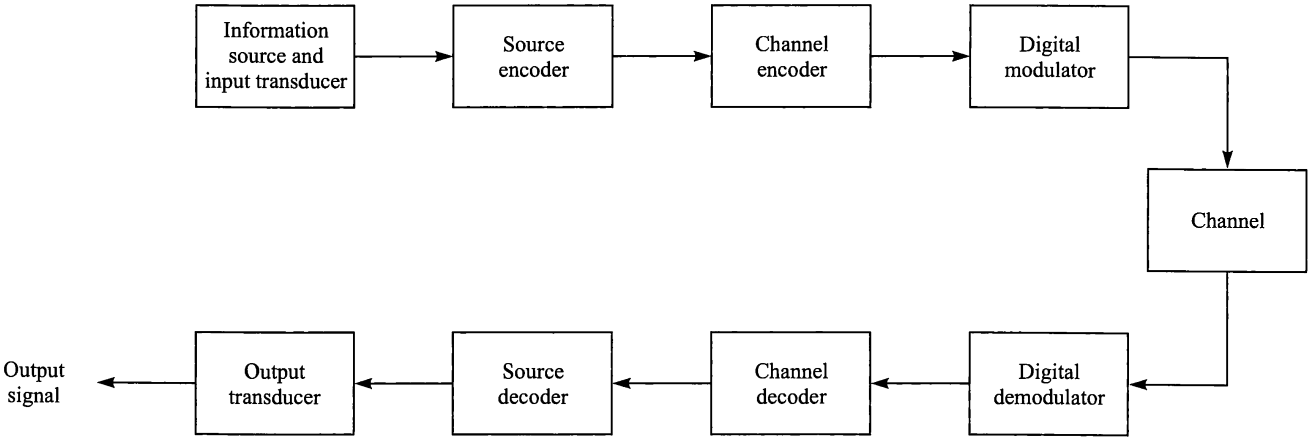


FIGURE 1.1–1
Basic elements of a digital communication system.

the fidelity of the received signal. In effect, redundancy in the information sequence aids the receiver in decoding the desired information sequence. For example, a (trivial) form of encoding of the binary information sequence is simply to repeat each binary digit m times, where m is some positive integer. More sophisticated (nontrivial) encoding involves taking k information bits at a time and mapping each k-bit sequence into a unique n-bit sequence, called a *code word.* The amount of redundancy introduced by encoding the data in this manner is measured by the ratio n/k. The reciprocal of this ratio, namely k/n, is called the rate of the code or, simply, the *code rate.*

The binary sequence at the output of the channel encoder is passed to the *digital modulator*, which serves as the interface to the communication channel. Since nearly all the communication channels encountered in practice are capable of transmitting electrical signals (waveforms), the primary purpose of the digital modulator is to map the binary information sequence into signal waveforms. To elaborate on this point, let us suppose that the coded information sequence is to be transmitted one bit at a time at some uniform rate R bits per second (bits/s). The digital modulator may simply map the binary digit 0 into a waveform $s_0(t)$ and the binary digit 1 into a waveform $s_1(t)$. In this manner, each bit from the channel encoder is transmitted separately. We call this *binary modulation.* Alternatively, the modulator may transmit b coded information bits at a time by using $M = 2^b$ distinct waveforms $s_i(t)$, $i = 0, 1, \ldots, M - 1$, one waveform for each of the 2^b possible b-bit sequences. We call this *M-ary modulation* ($M > 2$). Note that a new b-bit sequence enters the modulator every b/R seconds. Hence, when the channel bit rate R is fixed, the amount of time available to transmit one of the M waveforms corresponding to a b-bit sequence is b times the time period in a system that uses binary modulation.

The *communication channel* is the physical medium that is used to send the signal from the transmitter to the receiver. In wireless transmission, the channel may be the atmosphere (free space). On the other hand, telephone channels usually employ a variety of physical media, including wire lines, optical fiber cables, and wireless (microwave radio). Whatever the physical medium used for transmission of the information, the essential feature is that the transmitted signal is corrupted in a random manner by a variety of possible mechanisms, such as additive *thermal noise* generated by electronic devices; man-made noise, e.g., automobile ignition noise; and atmospheric noise, e.g., electrical lightning discharges during thunderstorms.

At the receiving end of a digital communication system, the *digital demodulator* processes the channel-corrupted transmitted waveform and reduces the waveforms to a sequence of numbers that represent estimates of the transmitted data symbols (binary or M-ary). This sequence of numbers is passed to the channel decoder, which attempts to reconstruct the original information sequence from knowledge of the code used by the channel encoder and the redundancy contained in the received data.

A measure of how well the demodulator and decoder perform is the frequency with which errors occur in the decoded sequence. More precisely, the average probability of a bit-error at the output of the decoder is a measure of the performance of the demodulator–decoder combination. In general, the probability of error is a function of the code characteristics, the types of waveforms used to transmit the information over the channel, the transmitter power, the characteristics of the channel (i.e., the amount of noise, the nature of the interference), and the method of demodulation and decoding. These items and their effect on performance will be discussed in detail in subsequent chapters.

As a final step, when an analog output is desired, the source decoder accepts the output sequence from the channel decoder and, from knowledge of the source encoding method used, attempts to reconstruct the original signal from the source. Because of channel decoding errors and possible distortion introduced by the source encoder, and perhaps, the source decoder, the signal at the output of the source decoder is an approximation to the original source output. The difference or some function of the difference between the original signal and the reconstructed signal is a measure of the distortion introduced by the digital communication system.

1.2

COMMUNICATION CHANNELS AND THEIR CHARACTERISTICS

As indicated in the preceding discussion, the communication channel provides the connection between the transmitter and the receiver. The physical channel may be a pair of wires that carry the electrical signal, or an optical fiber that carries the information on a modulated light beam, or an underwater ocean channel in which the information is transmitted acoustically, or free space over which the information-bearing signal is radiated by use of an antenna. Other media that can be characterized as communication channels are data storage media, such as magnetic tape, magnetic disks, and optical disks.

One common problem in signal transmission through any channel is additive noise. In general, additive noise is generated internally by components such as resistors and solid-state devices used to implement the communication system. This is sometimes called *thermal noise*. Other sources of noise and interference may arise externally to the system, such as interference from other users of the channel. When such noise and interference occupy the same frequency band as the desired signal, their effect can be minimized by the proper design of the transmitted signal and its demodulator at the receiver. Other types of signal degradations that may be encountered in transmission over the channel are signal attenuation, amplitude and phase distortion, and multipath distortion.

The effects of noise may be minimized by increasing the power in the transmitted signal. However, equipment and other practical constraints limit the power level in the transmitted signal. Another basic limitation is the available channel bandwidth. A bandwidth constraint is usually due to the physical limitations of the medium and the electronic components used to implement the transmitter and the receiver. These two limitations constrain the amount of data that can be transmitted reliably over any communication channel as we shall observe in later chapters. Below, we describe some of the important characteristics of several communication channels.

Wireline Channels

The telephone network makes extensive use of wire lines for voice signal transmission, as well as data and video transmission. Twisted-pair wire lines and coaxial cable are basically guided electromagnetic channels that provide relatively modest bandwidths. Telephone wire generally used to connect a customer to a central office has a bandwidth of several hundred kilohertz (kHz). On the other hand, coaxial cable has a usable bandwidth of several megahertz (MHz). Figure 1.2–1 illustrates the frequency range of guided electromagnetic channels, which include waveguides and optical fibers.

Signals transmitted through such channels are distorted in both amplitude and phase and further corrupted by additive noise. Twisted-pair wireline channels are also prone to crosstalk interference from physically adjacent channels. Because wireline channels carry a large percentage of our daily communications around the country and the world, much research has been performed on the characterization of their transmission properties and on methods for mitigating the amplitude and phase distortion encountered in signal transmission. In Chapter 9, we describe methods for designing optimum transmitted signals and their demodulation; in Chapter 10, we consider the design of channel equalizers that compensate for amplitude and phase distortion on these channels.

Fiber-Optic Channels

Optical fibers offer the communication system designer a channel bandwidth that is several orders of magnitude larger than coaxial cable channels. During the past two decades, optical fiber cables have been developed that have a relatively low signal attenuation, and highly reliable photonic devices have been developed for signal generation and signal detection. These technological advances have resulted in a rapid deployment of optical fiber channels, both in domestic telecommunication systems as well as for transcontinental communication. With the large bandwidth available on fiber-optic channels, it is possible for telephone companies to offer subscribers a wide array of telecommunication services, including voice, data, facsimile, and video.

The transmitter or modulator in a fiber-optic communication system is a light source, either a light-emitting diode (LED) or a laser. Information is transmitted by varying (modulating) the intensity of the light source with the message signal. The light propagates through the fiber as a light wave and is amplified periodically (in the case of

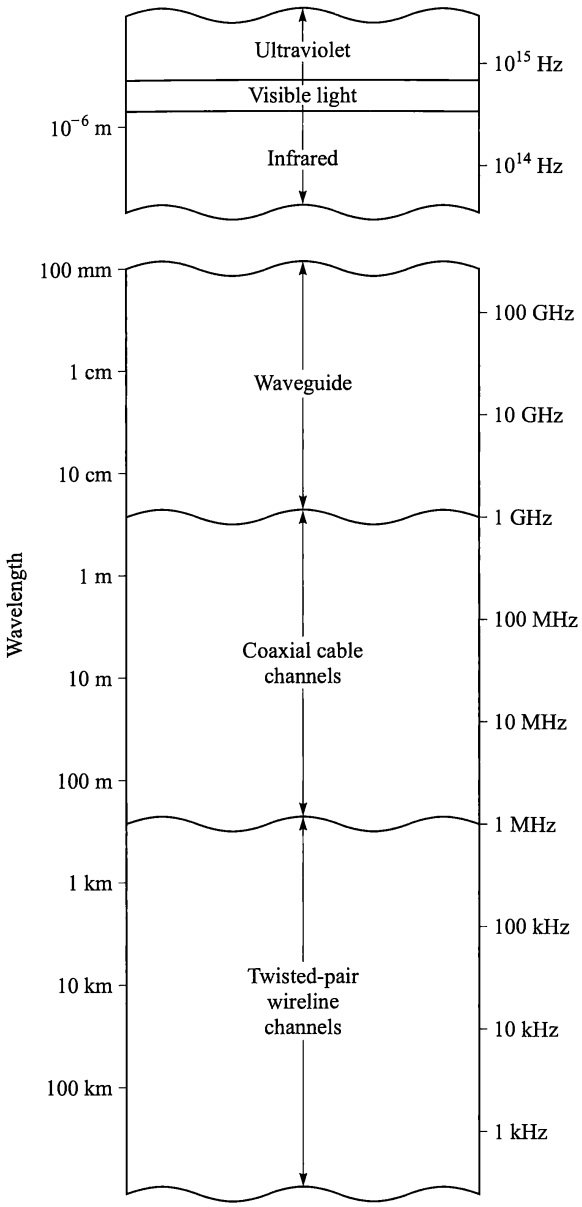

Ultraviolet
Visible light
Infrared
Waveguide
Coaxial cable channels
Twisted-pair wireline channels
Wavelength
10^{-6} m
100 mm
1 cm
10 cm
1 m
10 m
100 m
1 km
10 km
100 km
Frequency
10^{15} Hz
10^{14} Hz
100 GHz
10 GHz
1 GHz
100 MHz
10 MHz
1 MHz
100 kHz
10 kHz
1 kHz

FIGURE 1.2–1
Frequency range for guided wire channel.

digital transmission, it is detected and regenerated by repeaters) along the transmission path to compensate for signal attenuation. At the receiver, the light intensity is detected by a photodiode, whose output is an electrical signal that varies in direct proportion to the power of the light impinging on the photodiode. Sources of noise in fiber-optic channels are photodiodes and electronic amplifiers.

Wireless Electromagnetic Channels

In wireless communication systems, electromagnetic energy is coupled to the propagation medium by an antenna which serves as the radiator. The physical size and the configuration of the antenna depend primarily on the frequency of operation. To obtain efficient radiation of electromagnetic energy, the antenna must be longer than

$\frac{1}{10}$ of the wavelength. Consequently, a radio station transmitting in the amplitude-modulated (AM) frequency band, say at $f_c = 1$ MHz [corresponding to a wavelength of $\lambda = c/f_c = 300$ meters (m)], requires an antenna of at least 30 m. Other important characteristics and attributes of antennas for wireless transmission are described in Chapter 4.

Figure 1.2–2 illustrates the various frequency bands of the electromagnetic spectrum. The mode of propagation of electromagnetic waves in the atmosphere and in

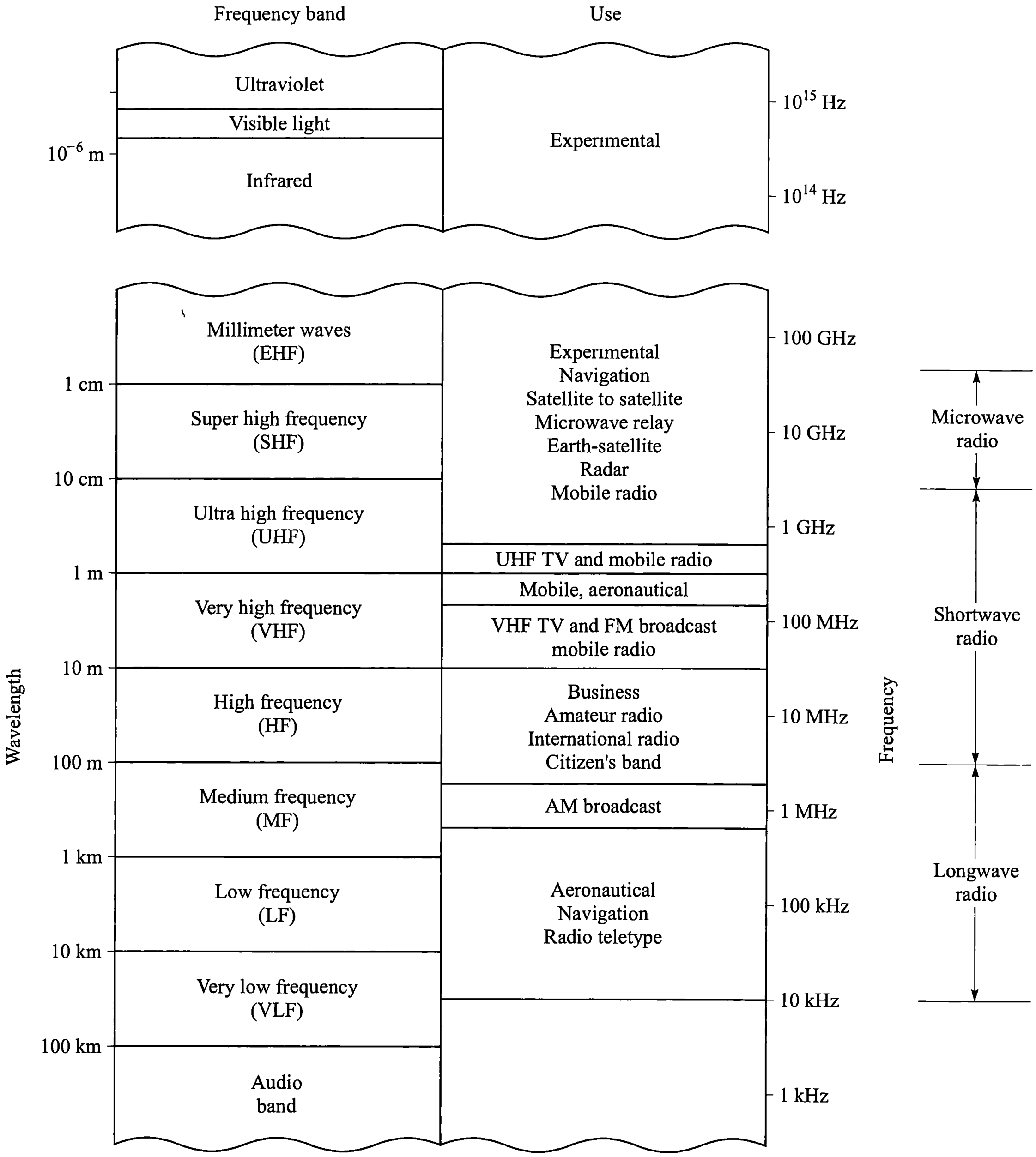

Frequency band
Use
Ultraviolet
Visible light
Infrared
Experimental
10^{-6} m
10^{15} Hz
10^{14} Hz
Millimeter waves (EHF)
Super high frequency (SHF)
Ultra high frequency (UHF)
Very high frequency (VHF)
High frequency (HF)
Medium frequency (MF)
Low frequency (LF)
Very low frequency (VLF)
Audio band
Experimental
Navigation
Satellite to satellite
Microwave relay
Earth-satellite
Radar
Mobile radio
UHF TV and mobile radio
Mobile, aeronautical
VHF TV and FM broadcast mobile radio
Business
Amateur radio
International radio
Citizen's band
AM broadcast
Aeronautical
Navigation
Radio teletype
Wavelength
1 cm
10 cm
1 m
10 m
100 m
1 km
10 km
100 km
Frequency
100 GHz
10 GHz
1 GHz
100 MHz
10 MHz
1 MHz
100 kHz
10 kHz
1 kHz
Microwave radio
Shortwave radio
Longwave radio

FIGURE 1.2–2

Frequency range for wireless electromagnetic channels. [*Adapted from Carlson* (1975), 2nd edition, © *McGraw-Hill Book Company Co. Reprinted with permission of the publisher.*]

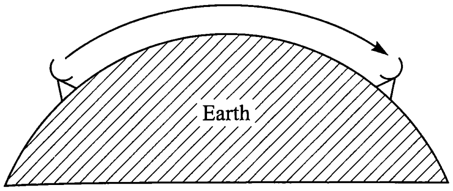


FIGURE 1.2–3
Illustration of ground-wave propagation.

free space may be subdivided into three categories, namely, ground-wave propagation, sky-wave propagation, and line-of-sight (LOS) propagation. In the very low frequency (VLF) and audio frequency bands, where the wavelengths exceed 10 km, the earth and the ionosphere act as a waveguide for electromagnetic wave propagation. In these frequency ranges, communication signals practically propagate around the globe. For this reason, these frequency bands are primarily used to provide navigational aids from shore to ships around the world. The channel bandwidths available in these frequency bands are relatively small (usually 1–10 percent of the center frequency), and hence the information that is transmitted through these channels is of relatively slow speed and generally confined to digital transmission. A dominant type of noise at these frequencies is generated from thunderstorm activity around the globe, especially in tropical regions. Interference results from the many users of these frequency bands.

Ground-wave propagation, as illustrated in Figure 1.2–3, is the dominant mode of propagation for frequencies in the medium frequency (MF) band (0.3–3 MHz). This is the frequency band used for AM broadcasting and maritime radio broadcasting. In AM broadcasting, the range with ground-wave propagation of even the more powerful radio stations is limited to about 150 km. Atmospheric noise, man-made noise, and thermal noise from electronic components at the receiver are dominant disturbances for signal transmission in the MF band.

Sky-wave propagation, as illustrated in Figure 1.2–4, results from transmitted signals being reflected (bent or refracted) from the ionosphere, which consists of several layers of charged particles ranging in altitude from 50 to 400 km above the surface of the earth. During the daytime hours, the heating of the lower atmosphere by the sun causes the formation of the lower layers at altitudes below 120 km. These lower layers, especially the D-layer, serve to absorb frequencies below 2 MHz, thus severely limiting sky-wave propagation of AM radio broadcast. However, during the nighttime hours, the electron density in the lower layers of the ionosphere drops sharply and the frequency absorption that occurs during the daytime is significantly reduced. As a consequence, powerful AM radio broadcast stations can propagate over large distances via sky wave over the F-layer of the ionosphere, which ranges from 140 to 400 km above the surface of the earth.

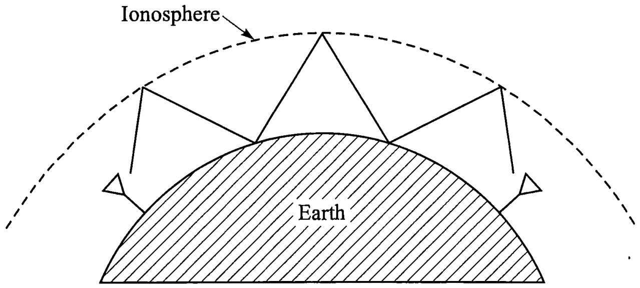


FIGURE 1.2–4
Illustration of sky-wave propagation.

A frequently occurring problem with electromagnetic wave propagation via sky wave in the high frequency (HF) range is *signal multipath*. Signal multipath occurs when the transmitted signal arrives at the receiver via multiple propagation paths at different delays. It generally results in intersymbol interference in a digital communication system. Moreover, the signal components arriving via different propagation paths may add destructively, resulting in a phenomenon called *signal fading*, which most people have experienced when listening to a distant radio station at night when sky wave is the dominant propagation mode. Additive noise in the HF range is a combination of atmospheric noise and thermal noise.

Sky-wave ionospheric propagation ceases to exist at frequencies above approximately 30 MHz, which is the end of the HF band. However, it is possible to have ionospheric scatter propagation at frequencies in the range 30–60 MHz, resulting from signal scattering from the lower ionosphere. It is also possible to communicate over distances of several hundred miles by use of tropospheric scattering at frequencies in the range 40–300 MHz. Troposcatter results from signal scattering due to particles in the atmosphere at altitudes of 10 miles or less. Generally, ionospheric scatter and tropospheric scatter involve large signal propagation losses and require a large amount of transmitter power and relatively large antennas.

Frequencies above 30 MHz propagate through the ionosphere with relatively little loss and make satellite and extraterrestrial communications possible. Hence, at frequencies in the very high frequency (VHF) band and higher, the dominant mode of electromagnetic propagation is LOS propagation. For terrestrial communication systems, this means that the transmitter and receiver antennas must be in direct LOS with relatively little or no obstruction. For this reason, television stations transmitting in the VHF and ultra high frequency (UHF) bands mount their antennas on high towers to achieve a broad coverage area.

In general, the coverage area for LOS propagation is limited by the curvature of the earth. If the transmitting antenna is mounted at a height h m above the surface of the earth, the distance to the radio horizon, assuming no physical obstructions such as mountains, is approximately $d = \sqrt{15h}$ km. For example, a television antenna mounted on a tower of 300 m in height provides a coverage of approximately 67 km. As another example, microwave radio relay systems used extensively for telephone and video transmission at frequencies above 1 gigahertz (GHz) have antennas mounted on tall towers or on the top of tall buildings.

The dominant noise limiting the performance of a communication system in VHF and UHF ranges is thermal noise generated in the receiver front end and cosmic noise picked up by the antenna. At frequencies in the super high frequency (SHF) band above 10 GHz, atmospheric conditions play a major role in signal propagation. For example, at 10 GHz, the attenuation ranges from about 0.003 decibel per kilometer (dB/km) in light rain to about 0.3 dB/km in heavy rain. At 100 GHz, the attenuation ranges from about 0.1 dB/km in light rain to about 6 dB/km in heavy rain. Hence, in this frequency range, heavy rain introduces extremely high propagation losses that can result in service outages (total breakdown in the communication system).

At frequencies above the extremely high frequency (EHF) band, we have the infrared and visible light regions of the electromagnetic spectrum, which can be used to provide LOS optical communication in free space. To date, these frequency bands have been used in experimental communication systems, such as satellite-to-satellite links.

Underwater Acoustic Channels

Over the past few decades, ocean exploration activity has been steadily increasing. Coupled with this increase is the need to transmit data, collected by sensors placed under water, to the surface of the ocean. From there, it is possible to relay the data via a satellite to a data collection center.

Electromagnetic waves do not propagate over long distances under water except at extremely low frequencies. However, the transmission of signals at such low frequencies is prohibitively expensive because of the large and powerful transmitters required. The attenuation of electromagnetic waves in water can be expressed in terms of the *skin depth*, which is the distance a signal is attenuated by $1/e$. For seawater, the skin depth $\delta = 250/\sqrt{f}$, where f is expressed in Hz and δ is in m. For example, at 10 kHz, the skin depth is 2.5 m. In contrast, acoustic signals propagate over distances of tens and even hundreds of kilometers.

An underwater acoustic channel is characterized as a multipath channel due to signal reflections from the surface and the bottom of the sea. Because of wave motion, the signal multipath components undergo time-varying propagation delays that result in signal fading. In addition, there is frequency-dependent attenuation, which is approximately proportional to the square of the signal frequency. The sound velocity is nominally about 1500 m/s, but the actual value will vary either above or below the nominal value depending on the depth at which the signal propagates.

Ambient ocean acoustic noise is caused by shrimp, fish, and various mammals. Near harbors, there is also man-made acoustic noise in addition to the ambient noise. In spite of this hostile environment, it is possible to design and implement efficient and highly reliable underwater acoustic communication systems for transmitting digital signals over large distances.

Storage Channels

Information storage and retrieval systems constitute a very significant part of data-handling activities on a daily basis. Magnetic tape, including digital audiotape and videotape, magnetic disks used for storing large amounts of computer data, optical disks used for computer data storage, and compact disks are examples of data storage systems that can be characterized as communication channels. The process of storing data on a magnetic tape or a magnetic or optical disk is equivalent to transmitting a signal over a telephone or a radio channel. The readback process and the signal processing involved in storage systems to recover the stored information are equivalent to the functions performed by a receiver in a telephone or radio communication system to recover the transmitted information.

Additive noise generated by the electronic components and interference from adjacent tracks is generally present in the readback signal of a storage system, just as is the case in a telephone or a radio communication system.

The amount of data that can be stored is generally limited by the size of the disk or tape and the density (number of bits stored per square inch) that can be achieved by the write/read electronic systems and heads. For example, a packing density of 10^9 bits per square inch has been demonstrated in magnetic disk storage systems. The speed at which data can be written on a disk or tape and the speed at which it can be read back are also limited by the associated mechanical and electrical subsystems that constitute an information storage system.

Channel coding and modulation are essential components of a well-designed digital magnetic or optical storage system. In the readback process, the signal is demodulated and the added redundancy introduced by the channel encoder is used to correct errors in the readback signal.

1.3

MATHEMATICAL MODELS FOR COMMUNICATION CHANNELS

In the design of communication systems for transmitting information through physical channels, we find it convenient to construct mathematical models that reflect the most important characteristics of the transmission medium. Then, the mathematical model for the channel is used in the design of the channel encoder and modulator at the transmitter and the demodulator and channel decoder at the receiver. Below, we provide a brief description of the channel models that are frequently used to characterize many of the physical channels that we encounter in practice.

The Additive Noise Channel

The simplest mathematical model for a communication channel is the additive noise channel, illustrated in Figure 1.3–1. In this model, the transmitted signal $s(t)$ is corrupted by an additive random noise process $n(t)$. Physically, the additive noise process may arise from electronic components and amplifiers at the receiver of the communication system or from interference encountered in transmission (as in the case of radio signal transmission).

If the noise is introduced primarily by electronic components and amplifiers at the receiver, it may be characterized as thermal noise. This type of noise is characterized statistically as a *Gaussian noise process*. Hence, the resulting mathematical model for the channel is usually called the *additive Gaussian noise channel*. Because this channel model applies to a broad class of physical communication channels and because of its mathematical tractability, this is the predominant channel model used in our communication system analysis and design. Channel attenuation is easily incorporated into the model. When the signal undergoes attenuation in transmission through the

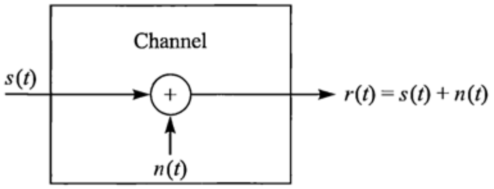


FIGURE 1.3–1
The additive noise channel.

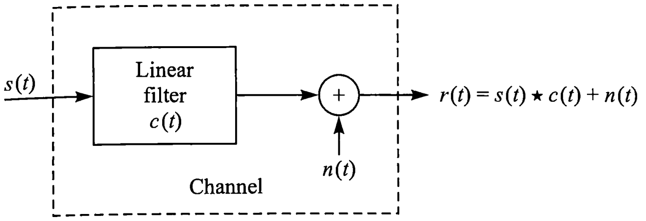


FIGURE 1.3–2
The linear filter channel with additive noise.

channel, the received signal is

$$r(t) = \alpha s(t) + n(t) \tag{1.3–1}$$

where α is the attenuation factor.

The Linear Filter Channel

In some physical channels, such as wireline telephone channels, filters are used to ensure that the transmitted signals do not exceed specified bandwidth limitations and thus do not interfere with one another. Such channels are generally characterized mathematically as linear filter channels with additive noise, as illustrated in Figure 1.3–2. Hence, if the channel input is the signal $s(t)$, the channel output is the signal

$$\begin{aligned} r(t) &= s(t) \star c(t) + n(t) \\ &= \int_{-\infty}^{\infty} c(\tau) s(t-\tau)\, d\tau + n(t) \end{aligned} \tag{1.3–2}$$

where $c(t)$ is the impulse response of the linear filter and $\star$ denotes convolution.

The Linear Time-Variant Filter Channel

Physical channels such as underwater acoustic channels and ionospheric radio channels that result in time-variant multipath propagation of the transmitted signal may be characterized mathematically as time-variant linear filters. Such linear filters are characterized by a time-variant channel impulse response $c(\tau; t)$, where $c(\tau; t)$ is the response of the channel at time t due to an impulse applied at time $t - \tau$. Thus, τ represents the "age" (elapsed-time) variable. The linear time-variant filter channel with additive noise is illustrated in Figure 1.3–3. For an input signal $s(t)$, the channel output signal is

$$\begin{aligned} r(t) &= s(t) \star c(\tau; t) + n(t) \\ &= \int_{-\infty}^{\infty} c(\tau; t) s(t-\tau)\, d\tau + n(t) \end{aligned} \tag{1.3–3}$$

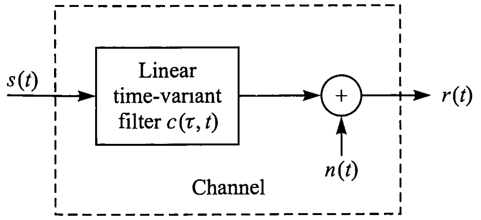


FIGURE 1.3–3
Linear time-variant filter channel with additive noise.

A good model for multipath signal propagation through physical channels, such as the ionosphere (at frequencies below 30 MHz) and mobile cellular radio channels, is a special case of (1.3–3) in which the time-variant impulse response has the form

$$c(\tau; t) = \sum_{k=1}^{L} a_k(t)\delta(\tau - \tau_k) \tag{1.3–4}$$

where the $\{a_k(t)\}$ represents the possibly time-variant attenuation factors for the L multipath propagation paths and $\{\tau_k\}$ are the corresponding time delays. If (1.3–4) is substituted into (1.3–3), the received signal has the form

$$r(t) = \sum_{k=1}^{L} a_k(t)s(t - \tau_k) + n(t) \tag{1.3–5}$$

Hence, the received signal consists of L multipath components, where the kth component is attenuated by $a_k(t)$ and delayed by τ_k.

The three mathematical models described above adequately characterize the great majority of the physical channels encountered in practice. These three channel models are used in this text for the analysis and design of communication systems.

1.4

A HISTORICAL PERSPECTIVE IN THE DEVELOPMENT OF DIGITAL COMMUNICATIONS

It is remarkable that the earliest form of electrical communication, namely *telegraphy*, was a digital communication system. The electric telegraph was developed by Samuel Morse and was demonstrated in 1837. Morse devised the variable-length binary code in which letters of the English alphabet are represented by a sequence of dots and dashes (code words). In this code, more frequently occurring letters are represented by short code words, while letters occurring less frequently are represented by longer code words. Thus, the *Morse code* was the precursor of the variable-length source coding methods described in Chapter 6.

Nearly 40 years later, in 1875, Emile Baudot devised a code for telegraphy in which every letter was encoded into fixed-length binary code words of length 5. In the *Baudot code*, binary code elements are of equal length and designated as mark and space.

Although Morse is responsible for the development of the first electrical digital communication system (telegraphy), the beginnings of what we now regard as modern digital communications stem from the work of Nyquist (1924), who investigated the problem of determining the maximum signaling rate that can be used over a telegraph channel of a given bandwidth without intersymbol interference. He formulated a model of a telegraph system in which a transmitted signal has the general form

$$s(t) = \sum_{n} a_n g(t - nT) \tag{1.4–1}$$

where $g(t)$ represents a basic pulse shape and $\{a_n\}$ is the binary data sequence of $\{\pm 1\}$ transmitted at a rate of $1/T$ bits/s. Nyquist set out to determine the optimum pulse shape that was band-limited to W Hz and maximized the bit rate under the constraint that the pulse caused no intersymbol interference at the sampling time k/T, $k = 0, \pm 1, \pm 2, \ldots$. His studies led him to conclude that the maximum pulse rate is $2W$ pulses/s. This rate is now called the *Nyquist rate*. Moreover, this pulse rate can be achieved by using the pulses $g(t) = (\sin 2\pi Wt)/2\pi Wt$. This pulse shape allows recovery of the data without intersymbol interference at the sampling instants. Nyquist's result is equivalent to a version of the sampling theorem for band-limited signals, which was later stated precisely by Shannon (1948b). The sampling theorem states that a signal of bandwidth W can be reconstructed from samples taken at the Nyquist rate of $2W$ samples/s using the interpolation formula

$$s(t) = \sum_n s\left(\frac{n}{2W}\right) \frac{\sin[2\pi W(t - n/2W)]}{2\pi W(t - n/2W)} \tag{1.4–2}$$

In light of Nyquist's work, Hartley (1928) considered the issue of the amount of data that can be transmitted reliably over a band-limited channel when multiple amplitude levels are used. Because of the presence of noise and other interference, Hartley postulated that the receiver can reliably estimate the received signal amplitude to some accuracy, say A_δ. This investigation led Hartley to conclude that there is a maximum data rate that can be communicated reliably over a band-limited channel when the maximum signal amplitude is limited to $A_{\max}$ (fixed power constraint) and the amplitude resolution is A_δ.

Another significant advance in the development of communications was the work of Kolmogorov (1939) and Wiener (1942), who considered the problem of estimating a desired signal waveform $s(t)$ in the presence of additive noise $n(t)$, based on observation of the received signal $r(t) = s(t) + n(t)$. This problem arises in signal demodulation. Kolmogorov and Wiener determined the linear filter whose output is the best mean-square approximation to the desired signal $s(t)$. The resulting filter is called the *optimum linear (Kolmogorov–Wiener) filter.*

Hartley's and Nyquist's results on the maximum transmission rate of digital information were precursors to the work of Shannon (1948a,b), who established the mathematical foundations for information transmission and derived the fundamental limits for digital communication systems. In his pioneering work, Shannon formulated the basic problem of reliable transmission of information in statistical terms, using probabilistic models for information sources and communication channels. Based on such a statistical formulation, he adopted a logarithmic measure for the information content of a source. He also demonstrated that the effect of a transmitter power constraint, a bandwidth constraint, and additive noise can be associated with the channel and incorporated into a single parameter, called the *channel capacity*. For example, in the case of an additive white (spectrally flat) Gaussian noise interference, an ideal band-limited channel of bandwidth W has a capacity C given by

$$C = W \log_2\left(1 + \frac{P}{WN_0}\right) \qquad \text{bits/s} \tag{1.4–3}$$

where P is the average transmitted power and N_0 is the power spectral density of the additive noise. The significance of the channel capacity is as follows: If the information rate R from the source is less than $C (R < C)$, then it is theoretically possible to achieve reliable (error-free) transmission through the channel by appropriate coding. On the other hand, if $R > C$, reliable transmission is not possible regardless of the amount of signal processing performed at the transmitter and receiver. Thus, Shannon established basic limits on communication of information and gave birth to a new field that is now called *information theory*.

Another important contribution to the field of digital communication is the work of Kotelnikov (1947), who provided a coherent analysis of the various digital communication systems based on a geometrical approach. Kotelnikov's approach was later expanded by Wozencraft and Jacobs (1965).

Following Shannon's publications came the classic work of Hamming (1950) on error-detecting and error-correcting codes to combat the detrimental effects of channel noise. Hamming's work stimulated many researchers in the years that followed, and a variety of new and powerful codes were discovered, many of which are used today in the implementation of modern communication systems.

The increase in demand for data transmission during the last four decades, coupled with the development of more sophisticated integrated circuits, has led to the development of very efficient and more reliable digital communication systems. In the course of these developments, Shannon's original results and the generalization of his results on maximum transmission limits over a channel and on bounds on the performance achieved have served as benchmarks for any given communication system design. The theoretical limits derived by Shannon and other researchers that contributed to the development of information theory serve as an ultimate goal in the continuing efforts to design and develop more efficient digital communication systems.

There have been many new advances in the area of digital communications following the early work of Shannon, Kotelnikov, and Hamming. Some of the most notable advances are the following:

- The development of new block codes by Muller (1954), Reed (1954), Reed and Solomon (1960), Bose and Ray-Chaudhuri (1960a,b), and Goppa (1970, 1971).
- The development of concatenated codes by Forney (1966a).
- The development of computationally efficient decoding of Bose–Chaudhuri-Hocquenghem (BCH) codes, e.g., the Berlekamp–Massey algorithm (see Chien, 1964; Berlekamp, 1968).
- The development of convolutional codes and decoding algorithms by Wozencraft and Reiffen (1961), Fano (1963), Zigangirov (1966), Jelinek (1969), Forney (1970b, 1972, 1974), and Viterbi (1967, 1971).
- The development of trellis-coded modulation by Ungerboeck (1982), Forney et al. (1984), Wei (1987), and others.
- The development of efficient source encodings algorithms for data compression, such as those devised by Ziv and Lempel (1977, 1978), and Linde et al. (1980).
- The development of low-density parity check (LDPC) codes and the sum-product decoding algorithm by Gallager (1963).
- The development of turbo codes and iterative decoding by Berrou et al. (1993).

1.5
OVERVIEW OF THE BOOK

Chapter 2 presents a review of deterministic and random signal analysis. Our primary objectives in this chapter are to review basic notions in the theory of probability and random variables and to establish some necessary notation.

Chapters 3 through 5 treat the geometric representation of various digital modulation signals, their demodulation, their error rate performance in additive, white Gaussian noise (AWGN) channels, and methods for synchronizing the receiver to the received signal waveforms.

Chapters 6 to 8 treat the topics of source coding, channel coding and decoding, and basic information theoretic limits on channel capacity, source information rates, and channel coding rates.

The design of efficient modulators and demodulators for linear filter channels with distortion is treated in Chapters 9 and 10. Channel equalization methods are described for mitigating the effects of channel distortion.

Chapter 11 is focused on multichannel and multicarrier communication systems, their efficient implementation, and their performance in AWGN channels.

Chapter 12 presents an introduction to direct sequence and frequency hopped spread spectrum signals and systems and an evaluation of their performance under worst-case interference conditions.

The design of signals and coding techniques for digital communication through fading multipath channels is the focus of Chapters 13 and 14. This material is especially relevant to the design and development of wireless communication systems.

Chapter 15 treats the use of multiple transmit and receive antennas for improving the performance of wireless communication systems through signal diversity and increasing the data rate via spatial multiplexing. The capacity of multiple antenna systems is evaluated and space-time codes are described for use in multiple antenna communication systems.

Chapter 16 of this book presents an introduction to multiuser communication systems and multiple access methods. We consider detection algorithms for uplink transmission in which multiple users transmit data to a common receiver (a base station) and evaluate their performance. We also present algorithms for suppressing multiple access interference in a broadcast communication system in which a transmitter employing multiple antennas transmits different data sequences simultaneously to different users.

1.6
BIBLIOGRAPHICAL NOTES AND REFERENCES

There are several historical treatments regarding the development of radio and telecommunications during the past century. These may be found in the books by McMahon (1984), Millman (1984), and Ryder and Fink (1984). We have already cited the classical works of Nyquist (1924), Hartley (1928), Kotelnikov (1947), Shannon (1948), and

Hamming (1950), as well as some of the more important advances that have occurred in the field since 1950. The collected papers by Shannon have been published by IEEE Press in a book edited by Sloane and Wyner (1993) and previously in Russia in a book edited by Dobrushin and Lupanov (1963). Other collected works published by the IEEE Press that might be of interest to the reader are *Key Papers in the Development of Coding Theory*, edited by Berlekamp (1974), and *Key Papers in the Development of Information Theory*, edited by Slepian (1974).

2

Deterministic and Random Signal Analysis

In this chapter we present the background material needed in the study of the following chapters. The analysis of deterministic and random signals and the study of different methods for their representation are the main topics of this chapter. In addition, we also introduce and study the main properties of some random variables frequently encountered in analysis of communication systems. We continue with a review of random processes, properties of lowpass and bandpass random processes, and series expansion of random processes.

Throughout this chapter, and the book, we assume that the reader is familiar with the properties of the Fourier transform as summarized in Table 2.0–1 and the important Fourier transform pairs given in Table 2.0–2.

In these tables we have used the following signal definitions.

$$\Pi(t) = \begin{cases} 1 & |t| < \frac{1}{2} \\ \frac{1}{2} & t = \pm\frac{1}{2} \\ 0 & \text{otherwise} \end{cases} \qquad \text{sinc}(t) = \begin{cases} \frac{\sin(\pi t)}{\pi t} & t \neq 0 \\ 1 & t = 0 \end{cases}$$

and

$$\text{sgn}(t) = \begin{cases} 1 & t > 0 \\ -1 & t < 0 \\ 0 & t = 0 \end{cases} \qquad \Lambda(t) = \Pi(t) \star \Pi(t) = \begin{cases} t+1 & -1 \le t < 0 \\ -t+1 & 0 \le t < 1 \\ 0 & \text{otherwise} \end{cases}$$

The unit step signal $u_{-1}(t)$ is defined as

$$u_{-1}(t) = \begin{cases} 1 & t > 0 \\ \frac{1}{2} & t = 0 \\ 0 & t < 0 \end{cases}$$

We also assume that the reader is familiar with elements of probability, random variables, and random processes as covered in standard texts such as Papoulis and Pillai (2002), Leon-Garcia (1994), and Stark and Woods (2002).

TABLE 2.0–1
Table of Fourier Transform Properties

Property	Signal	Fourier Transform
Linearity	$\alpha x_1(t) + \beta x_2(t)$	$\alpha X_1(f) + \beta X_2(f)$
Duality	$X(t)$	$x(-f)$
Conjugacy	$x^*(t)$	$X^*(-f)$
Time-scaling ($a \neq 0$)	$x(at)$	$\frac{1}{\lvert a\rvert} X\left(\frac{f}{a}\right)$
Time-shift	$x(t - t_0)$	$e^{-j2\pi f t_0} X(f)$
Modulation	$e^{j2\pi f_0 t} x(t)$	$X(f - f_0)$
Convolution	$x(t) \star y(t)$	$X(f)Y(f)$
Multiplication	$x(t)y(t)$	$X(f) \star Y(f)$
Differentiation	$\frac{d^n}{dt^n} x(t)$	$(j2\pi f)^n X(f)$
Differentiation in frequency	$t^n x(t)$	$\left(\frac{j}{2\pi}\right)^n \frac{d^n}{df^n} X(f)$
Integration	$\int_{-\infty}^{t} x(\tau)\, d\tau$	$\frac{X(f)}{j2\pi f} + \frac{1}{2} X(0)\delta(f)$
Parseval's theorem	$\int_{-\infty}^{\infty} x(t)y^*(t)\, dt =$	$\int_{-\infty}^{\infty} X(f)Y^*(f)\, df$
Rayleigh's theorem	$\int_{-\infty}^{\infty} \lvert x(t)\rvert^2\, dt =$	$\int_{-\infty}^{\infty} \lvert X(f)\rvert^2\, df$

2.1
BANDPASS AND LOWPASS SIGNAL REPRESENTATION

As was discussed in Chap. 1, the process of communication consists of transmission of the output of an information source over a communication channel. In almost all cases, the spectral characteristics of the information sequence do not directly match the spectral characteristics of the communication channel, and hence the information signal cannot be directly transmitted over the channel. In many cases the information signal is a low frequency (baseband) signal, and the available spectrum of the communication channel is at higher frequencies. Therefore, at the transmitter the information signal is translated to a higher frequency signal that matches the properties of the communication channel. This is the modulation process in which the baseband information signal is turned into a bandpass modulated signal. In this section we study the main properties of baseband and bandpass signals.

2.1–1 Bandpass and Lowpass Signals

In this section we will show that any real, narrowband, and high frequency signal—called a bandpass signal—can be represented in terms of a complex low frequency

TABLE 2.0–2
Table of Fourier Transform Pairs

Time Domain	Frequency Domain
$\delta(t)$	1
1	$\delta(f)$
$\delta(t-t_0)$	$e^{-j2\pi f t_0}$
$e^{j2\pi f_0 t}$	$\delta(f-f_0)$
$\cos(2\pi f_0 t)$	$\frac{1}{2}\delta(f-f_0)+\frac{1}{2}\delta(f+f_0)$
$\sin(2\pi f_0 t)$	$\frac{1}{2j}\delta(f-f_0)-\frac{1}{2j}\delta(f+f_0)$
$\Pi(t)$	$\operatorname{sinc}(f)$
$\operatorname{sinc}(t)$	$\Pi(f)$
$\Lambda(t)$	$\operatorname{sinc}^2(f)$
$\operatorname{sinc}^2(t)$	$\Lambda(f)$
$e^{-\alpha t}u_{-1}(t), \alpha > 0$	$\frac{1}{\alpha+j2\pi f}$
$te^{-\alpha t}u_{-1}(t), \alpha > 0$	$\frac{1}{(\alpha+j2\pi f)^2}$
$e^{-\alpha\lvert t\rvert}(\alpha > 0)$	$\frac{2\alpha}{\alpha^2+(2\pi f)^2}$
$e^{-\pi t^2}$	$e^{-\pi f^2}$
$\operatorname{sgn}(t)$	$\frac{1}{j\pi f}$
$u_{-1}(t)$	$\frac{1}{2}\delta(f)+\frac{1}{j2\pi f}$
$\frac{1}{2}\delta(t)+j\frac{1}{2\pi t}$	$u_{-1}(f)$
$\delta'(t)$	$j2\pi f$
$\delta^{(n)}(t)$	$(j2\pi f)^n$
$\frac{1}{t}$	$-j\pi \operatorname{sgn}(f)$
$\sum_{n=-\infty}^{\infty}\delta(t-nT_0)$	$\frac{1}{T_0}\sum_{n=-\infty}^{\infty}\delta\left(f-\frac{n}{T_0}\right)$

signal, called the lowpass equivalent of the original bandpass signal. This result makes it possible to work with the lowpass equivalents of bandpass signals instead of directly working with them, thus greatly simplifying the handling of bandpass signals. That is so because applying signal processing algorithms to lowpass signals is much easier due to lower required sampling rates which in turn result in lower rates of the sampled data.

The Fourier transform of a signal provides information about the frequency content, or *spectrum*, of the signal. The Fourier transform of a real signal $x(t)$ has *Hermitian symmetry*, i.e., $X(-f) = X^*(f)$, from which we conclude that $|X(-f)| = |X(f)|$ and $\angle X^*(f) = -\angle X(f)$. In other words, for real $x(t)$, the magnitude of $X(f)$ is even and

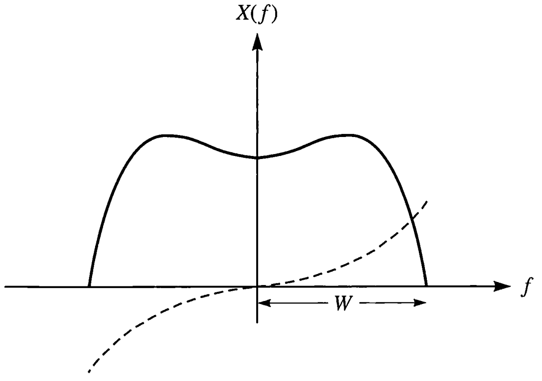


FIGURE 2.1–1
The spectrum of a real-valued lowpass (baseband) signal.

its phase is odd. Because of this symmetry, all information about the signal is in the positive (or negative) frequencies, and in particular $x(t)$ can be perfectly reconstructed by specifying $X(f)$ for $f \geq 0$. Based on this observation, for a real signal $x(t)$, we define the *bandwidth* as the smallest range of *positive frequencies* such that $X(f) = 0$ when $|f|$ is outside this range. It is clear that the bandwidth of a real signal is one-half of its frequency support set.

A *lowpass*, or *baseband*, signal is a signal whose spectrum is located around the zero frequency. For instance, speech, music, and video signals are all lowpass signals, although they have different spectral characteristics and bandwidths. Usually lowpass signals are low frequency signals, which means that in the time domain, they are slowly varying signals with no jumps or sudden variations. The bandwidth of a real lowpass signal is the minimum positive W such that $X(f) = 0$ outside $[-W, +W]$. For these signals the *frequency support*, i.e., the range of frequencies for which $X(f) \neq 0$, is $[-W, +W]$. An example of the spectrum of a real-valued lowpass signal is shown in Fig. 2.1–1. The solid line shows the magnitude spectrum $|X(f)|$, and the dashed line indicates the phase spectrum $\measuredangle X(f)$.

We also define the *positive spectrum* and the *negative spectrum* of a signal $x(t)$ as

$$X_+(f) = \begin{cases} X(f) & f > 0 \\ \frac{1}{2}X(0) & f = 0 \\ 0 & f < 0 \end{cases} \qquad X_-(f) = \begin{cases} X(f) & f < 0 \\ \frac{1}{2}X(0) & f = 0 \\ 0 & f > 0 \end{cases} \qquad (2.1\text{–}1)$$

It is clear that $X_+(f) = X(f)u_{-1}(f)$, $X_-(f) = X(f)u_{-1}(-f)$ and $X(f) = X_+(f) + X_-(f)$. For a real signal $x(t)$, since $X(f)$ is Hermitian, we have $X_-(f) = X_+^*(-f)$.

For a complex signal $x(t)$, the spectrum $X(f)$ is not symmetric; hence, the signal cannot be reconstructed from the information in the positive frequencies only. For complex signals, we define the *bandwidth* as one-half of the entire range of frequencies over which the spectrum is nonzero, i.e., one-half of the *frequency support* of the signal. This definition is for consistency with the definition of bandwidth for real signals. With this definition we can state that in general and for all signals, real or complex, the bandwidth is defined as one-half of the frequency support.

In practice, the spectral characteristics of the message signal and the communication channel do not always match, and it is required that the message signal be *modulated* by one of the many different modulation methods to match its spectral characteristics to

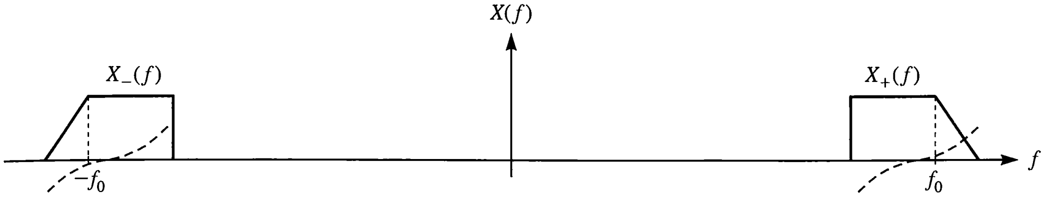


FIGURE 2.1–2
The spectrum of a real-valued bandpass signal.

the spectral characteristics of the channel. In this process, the spectrum of the lowpass message signal is translated to higher frequencies. The resulting modulated signal is a bandpass signal.

A *bandpass signal* is a real signal whose frequency content, or spectrum, is located around some frequency $\pm f_0$ which is far from zero. More formally, we define a bandpass signal to be a real signal $x(t)$ for which there exists positive f_0 and W such that the positive spectrum of $X(f)$, i.e., $X_+(f)$, is nonzero only in the interval $[f_0 - W/2, f_0 + W/2]$, where $W/2 < f_0$ (in practice, usually $W \ll f_0$). The frequency f_0 is called the *central frequency*. Obviously, the bandwidth of $x(t)$ is at most equal to W. Bandpass signals are usually high frequency signals which are characterized by rapid variations in the time domain.

An example of the spectrum of a bandpass signal is shown in Figure 2.1–2. Note that since the signal $x(t)$ is real, its magnitude spectrum (solid line) is even, and its phase spectrum (dashed line) is odd. Also, note that the central frequency f_0 is not necessarily the midband frequency of the bandpass signal. Due to the symmetry of the spectrum, $X_+(f)$ has all the information that is necessary to reconstruct $X(f)$. In fact we can write

$$X(f) = X_+(f) + X_-(f) = X_+(f) + X_+^*(-f) \tag{2.1–2}$$

which means that knowledge of $X_+(f)$ is sufficient to reconstruct $X(f)$.

2.1–2 Lowpass Equivalent of Bandpass Signals

We start by defining the *analytic signal*, or the *pre-envelope*, corresponding to $x(t)$ as the signal $x_+(t)$ whose Fourier transform is $X_+(f)$. This signal contains only positive frequency components, and its spectrum is not Hermitian. Therefore, in general, $x_+(t)$ is a complex signal. We have

$$\begin{aligned} x_+(t) &= \mathscr{F}^{-1}\left[X_+(f)\right] \\ &= \mathscr{F}^{-1}\left[X(f)u_{-1}(f)\right] \\ &= x(t) \star \left(\frac{1}{2}\delta(t) + j\frac{1}{2\pi t}\right) \\ &= \frac{1}{2}x(t) + \frac{j}{2}\widehat{x}(t) \end{aligned} \tag{2.1–3}$$

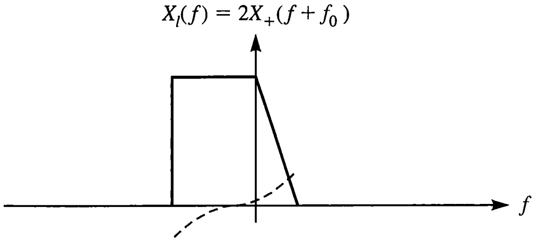


FIGURE 2.1–3
The spectrum of the lowpass equivalent of the signal shown in Figure 2.1–2.

where $\hat{x}(t) = \frac{1}{\pi t} \star x(t)$ is the *Hilbert transform* of $x(t)$. The Hilbert transform of $x(t)$ is obtained by introducing a phase shift of $-\frac{\pi}{2}$ at positive frequency components of $x(t)$ and $\frac{\pi}{2}$ at negative frequencies. In the frequency domain we have

$$\mathscr{F}\left[\hat{x}(t)\right] = -j\,\mathrm{sgn}(f)X(f) \tag{2.1–4}$$

Some of the properties of the Hilbert transform will be covered in the problems at the end of this chapter.

Now we define $x_l(t)$, the *lowpass equivalent*, or the *complex envelope*, of $x(t)$, as the signal whose spectrum is given by $2X_+(f + f_0)$, i.e.,

$$X_l(f) = 2X_+(f + f_0) = 2X(f + f_0)u_{-1}(f + f_0) \tag{2.1–5}$$

Obviously the spectrum of $x_l(t)$ is located around the zero frequency, and therefore it is in general a complex lowpass signal. This signal is called the *lowpass equivalent* or the *complex envelope* of $x(t)$. The spectrum of the lowpass equivalent of the signal shown in Figure 2.1–2 is shown in Figure 2.1–3.

Applying the modulation theorem of the Fourier transform, we obtain

$$\begin{aligned} x_l(t) &= \mathscr{F}^{-1}[X_l(f)] \\ &= 2x_+(t)e^{-j2\pi f_0 t} \\ &= (x(t) + j\hat{x}(t))e^{-j2\pi f_0 t} && \text{(2.1–6)} \\ &= (x(t)\cos 2\pi f_0 t + \hat{x}(t)\sin 2\pi f_0 t) \\ &\quad + j(\hat{x}(t)\cos 2\pi f_0 t - x(t)\sin 2\pi f_0 t) && \text{(2.1–7)} \end{aligned}$$

From Equation 2.1–6 we can write

$$x(t) = \mathrm{Re}\left[x_l(t)e^{j2\pi f_0 t}\right] \tag{2.1–8}$$

This relation expresses any bandpass signals in terms of its lowpass equivalent. Using Equations 2.1–2 and 2.1–5, we can write

$$X(f) = \frac{1}{2}\left[X_l(f - f_0) + X_l^*(-f - f_0)\right] \tag{2.1–9}$$

Equations 2.1–8, 2.1–9, 2.1–5, and 2.1–7 express $x(t)$ and $x_l(t)$ in terms of each other in the time and frequency domains.

The real and imaginary parts of $x_l(t)$ are called the *in-phase component* and the *quadrature component* of $x(t)$, respectively, and are denoted by $x_i(t)$ and $x_q(t)$. Both $x_i(t)$ and $x_q(t)$ are real-valued lowpass signals, and we have

$$x_l(t) = x_i(t) + jx_q(t) \tag{2.1–10}$$

Comparing Equations 2.1–10 and 2.1–7, we conclude that

$$\begin{aligned} x_i(t) &= x(t)\cos 2\pi f_0 t + \hat{x}(t)\sin 2\pi f_0 t \\ x_q(t) &= \hat{x}(t)\cos 2\pi f_0 t - x(t)\sin 2\pi f_0 t \end{aligned} \tag{2.1–11}$$

Solving Equation 2.1–11 for $x(t)$ and $\hat{x}(t)$ gives

$$\begin{aligned} x(t) &= x_i(t)\cos 2\pi f_0 t - x_q(t)\sin 2\pi f_0 t \\ \hat{x}(t) &= x_q(t)\cos 2\pi f_0 t + x_i(t)\sin 2\pi f_0 t \end{aligned} \tag{2.1–12}$$

Equation 2.1–12 shows that any bandpass signal $x(t)$ can be expressed in terms of two lowpass signals, namely, its in-phase and quadrature components.

Equation 2.1–10 expresses $x_l(t)$ in terms of its real and complex parts. We can write a similar relation in polar coordinates expressing $x(t)$ in terms of its magnitude and phase. If we define the *envelope* and *phase* of $x(t)$, denoted by $r_x(t)$ and $\theta_x(t)$, respectively, by

$$r_x(t) = \sqrt{x_i^2(t) + x_q^2(t)} \tag{2.1–13}$$

$$\theta_x(t) = \arctan \frac{x_q(t)}{x_i(t)} \tag{2.1–14}$$

we have

$$x_l(t) = r_x(t)e^{j\theta_x(t)} \tag{2.1–15}$$

Substituting this result into Equation 2.1–8 gives

$$x(t) = \text{Re}\left[r_x(t)e^{j(2\pi f_0 t + \theta_x(t))}\right] \tag{2.1–16}$$

resulting in

$$x(t) = r_x(t)\cos\left(2\pi f_0 t + \theta_x(t)\right) \tag{2.1–17}$$

A bandpass signal and its envelope are shown in Figure 2.1–4.

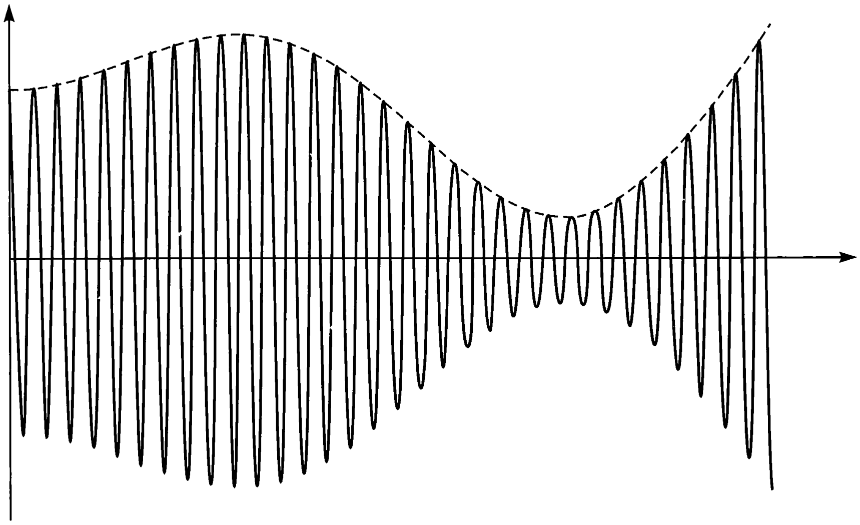


FIGURE 2.1–4
A bandpass signal. The dashed curve denotes the envelope.

It is important to note that $x_l(t)$—and consequently $x_i(t)$, $x_q(t)$, $r_x(t)$, and $\theta_x(t)$—depends on the choice of the central frequency f_0. For a given bandpass signal $x(t)$, different values of f_0—as long as $X_+(f)$ is nonzero only in the interval $[f_0 - W/2, f_0 + W/2]$, where $W/2 < f_0$—yield different lowpass signals $x_l(t)$. Therefore, it makes more sense to define the lowpass equivalent of a bandpass signal with respect to a specific f_0. Since in most cases the choice of f_0 is clear, we usually do not make this distinction.

Equations 2.1–12 and 2.1–17 provide two methods for representing a bandpass signal $x(t)$ in terms of two lowpass signals, one in terms of the in-phase and quadrature components and one in terms of the envelope and the phase. The two relations given in Equations 2.1–8 and 2.1–12 that express the bandpass signal in terms of the lowpass component(s) define the modulation process, i.e., the process of going from lowpass to bandpass. The system that implements this process is called a *modulator*. The structure of a general modulator implementing Equations 2.1–8 and 2.1–12 is shown in Figure 2.1–5(a) and (b). In this figure double lines and double blocks indicate complex values and operations.

Similarly, Equations 2.1–7 and 2.1–11 represent how $x_l(t)$, or $x_i(t)$ and $x_q(t)$, can be obtained from the bandpass signal $x(t)$. This process, i.e., extracting the lowpass signal from the bandpass signal, is called the *demodulation* process and is shown in Figure 2.1–6(a) and (b). In these block diagrams the block denoted by $\mathscr{H}$ represents a Hilbert transform, i.e., an LTI system with impulse response $h(t) = \frac{1}{\pi t}$ and transfer function $H(f) = -j\,\mathrm{sgn}(f)$.

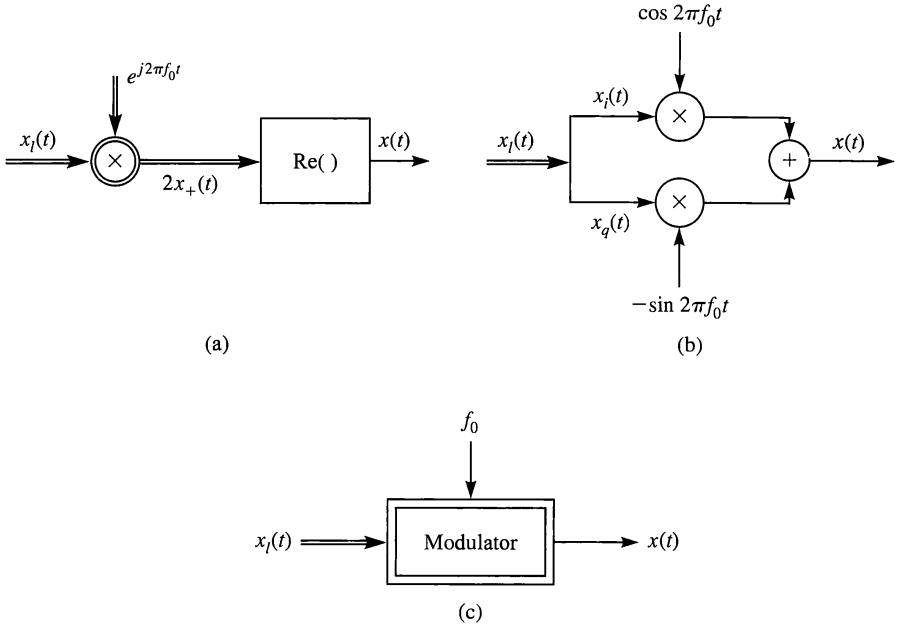
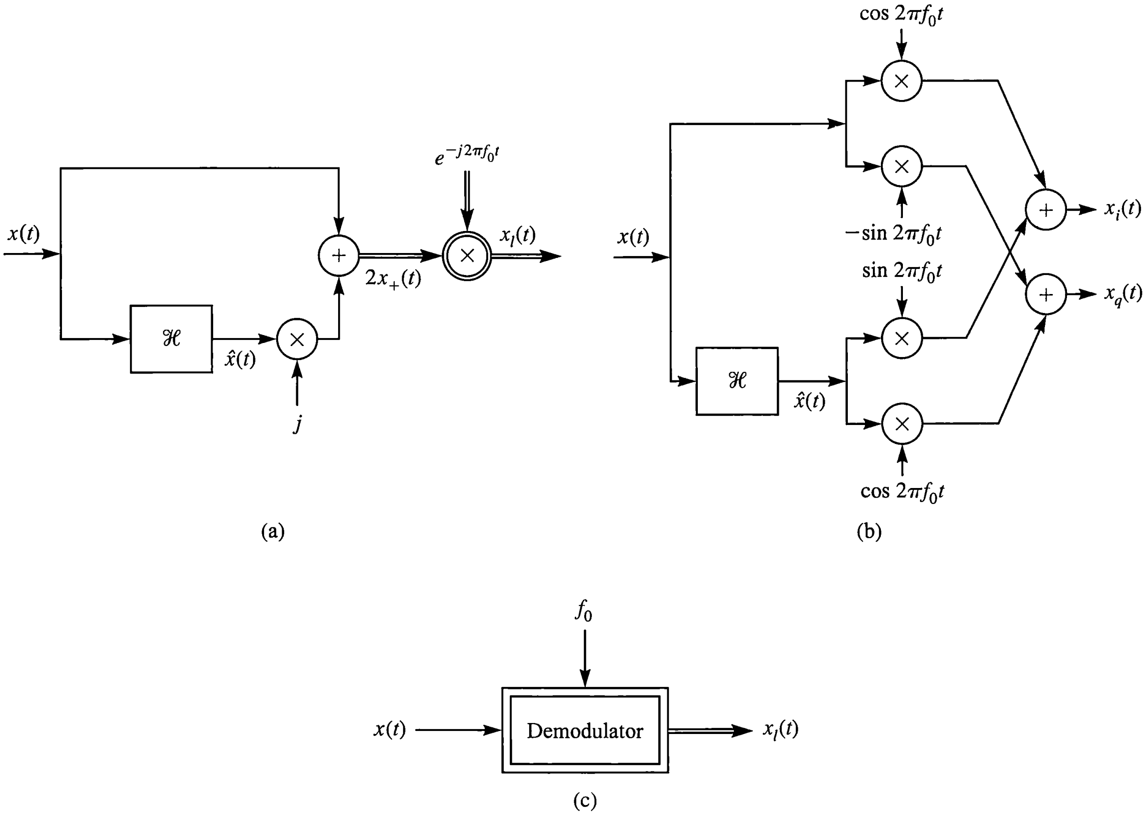


FIGURE 2.1–5

A complex (a) and real (b) modulator. A general representation for a modulator is shown in (c).


x(t)
ℋ
x̂(t)
j
+
2x₊(t)
e^{−j2πf₀t}
×
x_l(t)
cos 2πf₀t
−sin 2πf₀t
sin 2πf₀t
cos 2πf₀t
x_i(t)
x_q(t)
f₀
Demodulator

(a) (b) (c)

FIGURE 2.1–6
A complex (a) and real (b) demodulator. A general representation for a demodulator is shown in (c).

2.1–3 Energy Considerations

In this section we study the relation between energy contents of the signals introduced in the preceding pages. The *energy* of a signal $x(t)$ is defined as

$$\mathcal{E}_x = \int_{-\infty}^{\infty} |x(t)|^2\,dt \tag{2.1–18}$$

and by Rayleigh's relation from Table 2.0–1 we can write

$$\mathcal{E}_x = \int_{-\infty}^{\infty} |x(t)|^2\,dt = \int_{-\infty}^{\infty} |X(f)|^2\,dt \tag{2.1–19}$$

Since there is no overlap between $X_+(f)$ and $X_-(f)$, we have $X_+(f)X_-(f) = 0$, and hence

$$\begin{aligned}
\mathcal{E}_x &= \int_{-\infty}^{\infty} |X_+(f) + X_-(f)|^2\,df \\
&= \int_{-\infty}^{\infty} |X_+(f)|^2\,df + \int_{-\infty}^{\infty} |X_-(f)|^2\,df \\
&= 2\int_{-\infty}^{\infty} |X_+(f)|^2\,df \\
&= 2\mathcal{E}_{x_+}
\end{aligned} \tag{2.1–20}$$

On the other hand,

$$\begin{aligned}\mathcal{E}_x &= 2\int_{-\infty}^{\infty} |X_+(f)|^2\,df \\ &= 2\int_{-\infty}^{\infty} \left|\frac{X_l(f)}{2}\right|^2 df \\ &= \frac{1}{2}\mathcal{E}_{x_l}\end{aligned} \tag{2.1–21}$$

This shows that the energy in the lowpass equivalent signal is twice the energy in the bandpass signal.

We define the *inner product of two signals* $x(t)$ and $y(t)$ as

$$\langle x(t), y(t)\rangle = \int_{-\infty}^{\infty} x(t)y^*(t)\,dt = \int_{-\infty}^{\infty} X(f)Y^*(f)\,df \tag{2.1–22}$$

where we have used Parseval's relation from Table 2.0–1. Obviously

$$\mathcal{E}_x = \langle x(t), x(t)\rangle \tag{2.1–23}$$

In Problem 2.2 we prove that if $x(t)$ and $y(t)$ are two bandpass signals with lowpass equivalents $x_l(t)$ and $y_l(t)$ with respect to the same f_0, then

$$\langle x(t), y(t)\rangle = \frac{1}{2}\mathrm{Re}\,[\langle x_l(t), y_l(t)\rangle] \tag{2.1–24}$$

The complex quantity $\rho_{x,y}$, called the *cross-correlation coefficient* of $x(t)$ and $y(t)$, is defined as

$$\rho_{x,y} = \frac{\langle x(t), y(t)\rangle}{\sqrt{\mathcal{E}_x\mathcal{E}_y}} \tag{2.1–25}$$

and represents the normalized inner product between two signals. From $\mathcal{E}_{x_l} = 2\mathcal{E}_x$ and Equation 2.1–24 we can conclude that if $x(t)$ and $y(t)$ are bandpass signals with the same f_0, then

$$\rho_{x,y} = \mathrm{Re}\,(\rho_{x_l,y_l}) \tag{2.1–26}$$

Two signals are *orthogonal* if their inner product (and subsequently, their ρ) is zero. Note that if $\rho_{x_l,y_l} = 0$, then using Equation 2.1–26, we have $\rho_{x,y} = 0$; but the converse is not necessarily true. In other words, *orthogonality in the baseband implies orthogonality in the pass band, but not vice versa.*

EXAMPLE 2.1–1. Assume that $m(t)$ is a real baseband signal with bandwidth W, and define two signals $x(t) = m(t)\cos 2\pi f_0 t$ and $y(t) = m(t)\sin 2\pi f_0 t$, where $f_0 > W$. Comparing these relations with Equation 2.1–12, we conclude that

$$\begin{aligned} x_i(t) &= m(t) \qquad & x_q(t) &= 0 \\ y_i(t) &= 0 \qquad & y_q(t) &= -m(t)\end{aligned}$$

or, equivalently,

$$\begin{aligned} x_l(t) &= m(t) \\ y_l(t) &= -jm(t) \end{aligned}$$

Note that here

$$\rho_{x_l,y_l} = j\int_{-\infty}^{\infty} m^2(t) = j\mathcal{E}_m$$

Therefore,

$$\rho_{x,y} = \mathrm{Re}\,(\rho_{x_l,y_l}) = \mathrm{Re}\,(j\mathcal{E}_m) = 0$$

This means that $x(t)$ and $y(t)$ are orthogonal, but their lowpass equivalents are not orthogonal.

2.1–4 Lowpass Equivalent of a Bandpass System

A bandpass system is a system whose transfer function is located around a frequency f_0 (and its mirror image $-f_0$). More formally, we define a bandpass system as a system whose impulse response $h(t)$ is a bandpass signal. Since $h(t)$ is bandpass, it has a lowpass equivalent denoted by $h_l(t)$ where

$$h(t) = \mathrm{Re}\left[h_l(t)e^{j2\pi f_0 t}\right] \tag{2.1–27}$$

If a bandpass signal $x(t)$ passes through a bandpass system with impulse response $h(t)$, then obviously the output will be a bandpass signal $y(t)$. The relation between the spectra of the input and the output is given by

$$Y(f) = X(f)H(f) \tag{2.1–28}$$

Using Equation 2.1–5, we have

$$\begin{aligned} Y_l(f) &= 2Y(f+f_0)u_{-1}(f+f_0) \\ &= 2X(f+f_0)H(f+f_0)u_{-1}(f+f_0) \\ &= \frac{1}{2}\left[2X(f+f_0)u_{-1}(f+f_0)\right]\left[2H(f+f_0)u_{-1}(f+f_0)\right] \\ &= \frac{1}{2}X_l(f)H_l(f) \end{aligned} \tag{2.1–29}$$

where we have used the fact that for $f > -f_0$, which is the range of frequencies of interest, $u_{-1}^2(f+f_0) = u_{-1}(f+f_0) = 1$. In the time domain we have

$$y_l(t) = \frac{1}{2}x_l(t) \star h_l(t) \tag{2.1–30}$$

Equations 2.1–29 and 2.1–30 show that when a bandpass signal passes through a bandpass system, the input-output relation between the lowpass equivalents is very similar to the relation between the bandpass signals, the only difference being that for the lowpass equivalents a factor of $\frac{1}{2}$ is introduced.

2.2
SIGNAL SPACE REPRESENTATION OF WAVEFORMS

Signal space (or vector) representation of signals is a very effective and useful tool in the analysis of digitally modulated signals. We cover this important approach in this section and show that any set of signals is equivalent to a set of vectors. We show that signals have the same basic properties of vectors. We study methods of determining an equivalent set of vectors for a set of signals and introduce the notion of signal space representation, or *signal constellation*, of a set of waveforms.

2.2–1 Vector Space Concepts

A vector $\boldsymbol{v}$ in an n-dimensional space is characterized by its n components $v_1\ v_2 \cdots v_n$. Let $\boldsymbol{v}$ denote a column vector, i.e., $\boldsymbol{v} = [v_1\ v_2 \cdots v_n]^t$, where $\boldsymbol{A}^t$ denotes the transpose of matrix $\boldsymbol{A}$. The *inner product* of two n-dimensional vectors $\boldsymbol{v}_1 = [v_{11}\ v_{12} \cdots v_{1n}]^t$ and $\boldsymbol{v}_2 = [v_{21}\ v_{22} \cdots v_{2n}]^t$ is defined as

$$\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle = \boldsymbol{v}_1 \cdot \boldsymbol{v}_2 = \sum_{i=1}^{n} v_{1i} v_{2i}^* = \boldsymbol{v}_2^H \boldsymbol{v}_1 \tag{2.2–1}$$

where $\boldsymbol{A}^H$ denotes the *Hermitian transpose* of the matrix $\boldsymbol{A}$, i.e., the result of first transposing the matrix and then conjugating its elements. From the definition of the inner product of two vectors it follows that

$$\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle = \langle \boldsymbol{v}_2, \boldsymbol{v}_1 \rangle^* \tag{2.2–2}$$

and therefore,

$$\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle + \langle \boldsymbol{v}_2, \boldsymbol{v}_1 \rangle = 2\,\mathrm{Re}\,[\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle] \tag{2.2–3}$$

A vector may also be represented as a linear combination of orthogonal unit vectors or an *orthonormal basis* $\boldsymbol{e}_i$, $1 \le i \le n$, i.e.,

$$\boldsymbol{v} = \sum_{i=1}^{n} v_i \boldsymbol{e}_i \tag{2.2–4}$$

where, by definition, a unit vector has length unity and v_i is the projection of the vector $\boldsymbol{v}$ onto the unit vector $\boldsymbol{e}_i$, i.e., $v_i = \langle \boldsymbol{v}, \boldsymbol{e}_i \rangle$. Two vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are *orthogonal* if $\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle = 0$. More generally, a set of m vectors $\boldsymbol{v}_k$, $1 \le k \le m$, are orthogonal if $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = 0$ for all $1 \le i, j \le m$, and $i \ne j$. The *norm* of a vector $\boldsymbol{v}$ is denoted by $\|\boldsymbol{v}\|$ and is defined as

$$\|\boldsymbol{v}\| = (\langle \boldsymbol{v}, \boldsymbol{v} \rangle)^{1/2} = \sqrt{\sum_{i=1}^{n} |v_i|^2} \tag{2.2–5}$$

which in the n-dimensional space is simply the length of the vector. A set of m vectors is said to be *orthonormal* if the vectors are orthogonal and each vector has a unit norm. A set of m vectors is said to be *linearly independent* if no one vector can be represented as a linear combination of the remaining vectors. Any two n-dimensional vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ satisfy the *triangle inequality*

$$\|\boldsymbol{v}_1 + \boldsymbol{v}_2\| \leq \|\boldsymbol{v}_1\| + \|\boldsymbol{v}_2\| \qquad (2.2\text{–}6)$$

with equality if $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are in the same direction, i.e., $\boldsymbol{v}_1 = a\boldsymbol{v}_2$ where a is a positive real scalar. The *Cauchy–Schwarz inequality* states that

$$|\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle| \leq \|\boldsymbol{v}_1\| \cdot \|\boldsymbol{v}_2\| \qquad (2.2\text{–}7)$$

with equality if $\boldsymbol{v}_1 = a\boldsymbol{v}_2$ for some complex scalar a. The norm square of the sum of two vectors may be expressed as

$$\|\boldsymbol{v}_1 + \boldsymbol{v}_2\|^2 = \|\boldsymbol{v}_1\|^2 + \|\boldsymbol{v}_2\|^2 + 2\,\mathrm{Re}\,[\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle] \qquad (2.2\text{–}8)$$

If $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are orthogonal, then $\langle \boldsymbol{v}_1, \boldsymbol{v}_2 \rangle = 0$ and, hence,

$$\|\boldsymbol{v}_1 + \boldsymbol{v}_2\|^2 = \|\boldsymbol{v}_1\|^2 + \|\boldsymbol{v}_2\|^2 \qquad (2.2\text{–}9)$$

This is the *Pythagorean relation* for two orthogonal n-dimensional vectors. From matrix algebra, we recall that a linear transformation in an n-dimensional vector space is a matrix transformation of the form $\boldsymbol{v}' = \boldsymbol{A}\boldsymbol{v}$, where the matrix $\boldsymbol{A}$ transforms the vector $\boldsymbol{v}$ into some vector $\boldsymbol{v}'$. In the special case where $\boldsymbol{v}' = \lambda\boldsymbol{v}$, i.e.,

$$\boldsymbol{A}\boldsymbol{v} = \lambda\boldsymbol{v}$$

where λ is some scalar, the vector $\boldsymbol{v}$ is called an *eigenvector* of the transformation and λ is the corresponding *eigenvalue.*

Finally, let us review the *Gram–Schmidt procedure* for constructing a set of orthonormal vectors from a set of n-dimensional vectors $\boldsymbol{v}_i$, $1 \leq i \leq m$. We begin by arbitrarily selecting a vector from the set, say, $\boldsymbol{v}_1$. By normalizing its length, we obtain the first vector, say,

$$\boldsymbol{u}_1 = \frac{\boldsymbol{v}_1}{\|\boldsymbol{v}_1\|} \qquad (2.2\text{–}10)$$

Next, we may select $\boldsymbol{v}_2$ and, first, subtract the projection of $\boldsymbol{v}_2$ onto $\boldsymbol{u}_1$. Thus, we obtain

$$\boldsymbol{u}_2' = \boldsymbol{v}_2 - (\langle \boldsymbol{v}_2, \boldsymbol{u}_1 \rangle)\boldsymbol{u}_1 \qquad (2.2\text{–}11)$$

Then we normalize the vector $\boldsymbol{u}_2'$ to unit length. This yields

$$\boldsymbol{u}_2 = \frac{\boldsymbol{u}_2'}{\|\boldsymbol{u}_2'\|} \qquad (2.2\text{–}12)$$

The procedure continues by selecting $\boldsymbol{v}_3$ and subtracting the projections of $\boldsymbol{v}_3$ into $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$. Thus, we have

$$\boldsymbol{u}_3' = \boldsymbol{v}_3 - (\langle \boldsymbol{v}_3, \boldsymbol{u}_1 \rangle)\boldsymbol{u}_1 - (\langle \boldsymbol{v}_3, \boldsymbol{u}_2 \rangle)\boldsymbol{u}_2 \qquad (2.2\text{–}13)$$

Then the orthonormal vector $\boldsymbol{u}_3$ is

$$\boldsymbol{u}_3 = \frac{\boldsymbol{u}_3'}{\|\boldsymbol{u}_3'\|} \qquad (2.2\text{–}14)$$

By continuing this procedure, we construct a set of N orthonormal vectors, where $N \le \min(m, n)$.

2.2–2 Signal Space Concepts

As in the case of vectors, we may develop a parallel treatment for a set of signals. The *inner product* of two generally complex-valued signals $x_1(t)$ and $x_2(t)$ is denoted by $\langle x_1(t), x_2(t)\rangle$ and defined as

$$\langle x_1(t), x_2(t)\rangle = \int_{-\infty}^{\infty} x_1(t)x_2^*(t)\,dt \tag{2.2–15}$$

similar to Equation 2.1–22. The signals are *orthogonal* if their inner product is zero. The *norm of a signal* is defined as

$$\|x(t)\| = \left(\int_{-\infty}^{\infty} |x(t)|^2\,dt\right)^{1/2} = \sqrt{\mathcal{E}_x} \tag{2.2–16}$$

where $\mathcal{E}_x$ is the energy in $x(t)$. A set of m signals is *orthonormal* if they are orthogonal and their norms are all unity. A set of m signals is *linearly independent* if no signal can be represented as a linear combination of the remaining signals. The *triangle inequality* for two signals is simply

$$\|x_1(t) + x_2(t)\| \le \|x_1(t)\| + \|\, x_2(t)\| \tag{2.2–17}$$

and the Cauchy–Schwarz inequality is

$$|\langle x_1(t), x_2(t)\rangle| \le \|x_1(t)\| \cdot \|x_2(t)\| = \sqrt{\mathcal{E}_{x_1}\mathcal{E}_{x_2}} \tag{2.2–18}$$

or, equivalently,

$$\left|\int_{-\infty}^{\infty} x_1(t)x_2^*(t)\,dt\right| \le \left|\int_{-\infty}^{\infty} |x_1(t)|^2\,dt\right|^{1/2} \left|\int_{-\infty}^{\infty} |x_2(t)|^2\,dt\right|^{1/2} \tag{2.2–19}$$

with equality when $x_2(t) = ax_1(t)$, where a is any complex number.

2.2–3 Orthogonal Expansions of Signals

In this section, we develop a vector representation for signal waveforms, and thus we demonstrate an equivalence between a signal waveform and its vector representation. Suppose that $s(t)$ is a deterministic signal with finite energy

$$\mathcal{E}_s = \int_{-\infty}^{\infty} |s(t)|^2\,dt \tag{2.2–20}$$

Furthermore, suppose that there exists a set of functions $\{\phi_n(t), n = 1, 2, \ldots, K\}$ that are orthonormal in the sense that

$$\langle \phi_n(t), \phi_m(t)\rangle = \int_{-\infty}^{\infty} \phi_n(t)\phi_m^*(t)\,dt = \begin{cases} 1 & m = n \\ 0 & m \ne n \end{cases} \tag{2.2–21}$$

We may approximate the signal $s(t)$ by a weighted linear combination of these functions, i.e.,

$$\widehat{s}(t) = \sum_{k=1}^{K} s_k \phi_k(t) \tag{2.2–22}$$

where $\{s_k, 1 \le k \le K\}$ are the coefficients in the approximation of $s(t)$. The approximation error incurred is

$$e(t) = s(t) - \widehat{s}(t)$$

Let us select the coefficients $\{s_k\}$ so as to minimize the energy $\mathcal{E}_e$ of the approximation error. Thus,

$$\mathcal{E}_e = \int_{-\infty}^{\infty} |s(t) - \widehat{s}(t)|^2 \, dt \tag{2.2–23}$$

$$= \int_{-\infty}^{\infty} \left| s(t) - \sum_{k=1}^{K} s_k \phi_k(t) \right|^2 dt \tag{2.2–24}$$

The optimum coefficients in the series expansion of $s(t)$ may be found by differentiating Equation 2.2–23 with respect to each of the coefficients $\{s_k\}$ and setting the first derivatives to zero. Alternatively, we may use a well-known result from estimation theory based on the mean square error criterion, which, simply stated, is that the minimum of $\mathcal{E}_e$ with respect to the $\{s_k\}$ is obtained when the error is orthogonal to each of the functions in the series expansion. Thus,

$$\int_{-\infty}^{\infty} \left[s(t) - \sum_{k=1}^{K} s_k \phi_k(t) \right] \phi_n^*(t) \, dt = 0, \qquad n = 1, 2, \ldots, K \tag{2.2–25}$$

Since the functions $\{\phi_n(t)\}$ are orthonormal, Equation 2.2–25 reduces to

$$s_n = \langle s(t), \phi_n(t) \rangle = \int_{-\infty}^{\infty} s(t) \phi_n^*(t) \, dt, \qquad n = 1, 2, \ldots, K \tag{2.2–26}$$

Thus, the coefficients are obtained by projecting the signal $s(t)$ onto each of the functions $\{\phi_n(t)\}$. Consequently, $\widehat{s}(t)$ is the projection of $s(t)$ onto the K-dimensional signal space spanned by the functions $\{\phi_n(t)\}$, and therefore it is orthogonal to the error signal $e(t) = s(t) - \widehat{s}(t)$, i.e., $\langle e(t), \widehat{s}(t) \rangle = 0$. The minimum mean-square approximation error is

$$\mathcal{E}_{\min} = \int_{-\infty}^{\infty} e(t) s^*(t) \, dt \tag{2.2–27}$$

$$= \int_{-\infty}^{\infty} |s(t)|^2 \, dt - \int_{-\infty}^{\infty} \sum_{k=1}^{K} s_k \phi_k(t) s^*(t) \, dt \tag{2.2–28}$$

$$= \mathcal{E}_s - \sum_{k=1}^{K} |s_k|^2 \tag{2.2–29}$$

which is nonnegative, by definition. When the minimum mean square approximation error $\mathcal{E}_{\min} = 0$,

$$\mathcal{E}_s = \sum_{k=1}^{K} |s_k|^2 = \int_{-\infty}^{\infty} |s(t)|^2 \, dt \tag{2.2–30}$$

Under the condition that $\mathcal{E}_{\min} = 0$, we may express $s(t)$ as

$$s(t) = \sum_{k=1}^{K} s_k \phi_k(t) \tag{2.2–31}$$

where it is understood that equality of $s(t)$ to its series expansion holds in the sense that the approximation error has zero energy.

When every finite energy signal can be represented by a series expansion of the form in Equation 2.2–31 for which $\mathcal{E}_{\min} = 0$, the set of orthonormal functions $\{\phi_n(t)\}$ is said to be *complete*.

EXAMPLE 2.2–1. TRIGONOMETRIC FOURIER SERIES: Consider a finite energy real signal $s(t)$ that is zero everywhere except in the range $0 \le t \le T$ and has a finite number of discontinuities in this interval. Its periodic extension can be represented in a Fourier series as

$$s(t) = \sum_{k=0}^{\infty} \left(a_k \cos \frac{2\pi k t}{T} + b_k \sin \frac{2\pi k t}{T} \right) \tag{2.2–32}$$

where the coefficients $\{a_k, b_k\}$ that minimize the mean square error are given by

$$\begin{aligned} a_0 &= \frac{1}{T} \int_0^T s(t) \, dt \\ a_k &= \frac{2}{T} \int_0^T s(t) \cos \frac{2\pi k t}{T} \, dt, \qquad k = 1, 2, 3, \ldots \\ b_k &= \frac{2}{T} \int_0^T s(t) \sin \frac{2\pi k t}{T} \, dt, \qquad k = 1, 2, 3, \ldots \end{aligned} \tag{2.2–33}$$

The set of functions $\{1/\sqrt{T}, \sqrt{2/T} \cos 2\pi k t/T, \sqrt{2/T} \sin 2\pi k t/T\}$ is a complete set for the expansion of periodic signals on the interval $[0, T]$, and, hence, the series expansion results in zero mean square error.

EXAMPLE 2.2–2. EXPONENTIAL FOURIER SERIES: Consider a general finite energy signal $s(t)$ (real or complex) that is zero everywhere except in the range $0 \le t \le T$ and has a finite number of discontinuities in this interval. Its periodic extension can be represented in an exponential Fourier series as

$$s(t) = \sum_{n=-\infty}^{\infty} x_n \, e^{j2\pi \frac{n}{T} t} \tag{2.2–34}$$

where the coefficients $\{x_n\}$ that minimize the mean square error are given by

$$x_n = \frac{1}{T} \int_{-\infty}^{\infty} x(t) e^{-j2\pi \frac{n}{T} t} dt \tag{2.2–35}$$

The set of functions $\{\sqrt{1/T}e^{j2\pi\frac{n}{T}t}\}$ is a complete set for expansion of periodic signals on the interval $[0, T]$, and, hence, the series expansion results in zero mean square error.

2.2–4 Gram-Schmidt Procedure

Now suppose that we have a set of finite energy signal waveforms $\{s_m(t), m = 1, 2, \ldots, M\}$ and we wish to construct a set of orthonormal waveforms. The *Gram-Schmidt orthogonalization procedure* allows us to construct such a set. This procedure is similar to the one described in Section 2.2–1 for vectors. We begin with the first waveform $s_1(t)$, which is assumed to have energy $\mathcal{E}_1$. The first orthonormal waveform is simply constructed as

$$\phi_1(t) = \frac{s_1(t)}{\sqrt{\mathcal{E}_1}} \tag{2.2–36}$$

Thus, $\phi_1(t)$ is simply $s_1(t)$ normalized to unit energy. The second waveform is constructed from $s_2(t)$ by first computing the projection of $s_2(t)$ onto $\phi_1(t)$, which is

$$c_{21} = \langle s_2(t), \phi_1(t)\rangle = \int_{-\infty}^{\infty} s_2(t)\phi_1^*(t)\,dt \tag{2.2–37}$$

Then $c_{21}\phi_1(t)$ is subtracted from $s_2(t)$ to yield

$$\gamma_2(t) = s_2(t) - c_{21}\phi_1(t) \tag{2.2–38}$$

This waveform is orthogonal to $\phi_1(t)$, but it does not have unit energy. If $\mathcal{E}_2$ denotes the energy of $\gamma_2(t)$, i.e.,

$$\mathcal{E}_2 = \int_{-\infty}^{\infty} \gamma_2^2(t)\,dt$$

the normalized waveform that is orthogonal to $\phi_1(t)$ is

$$\phi_2(t) = \frac{\gamma_2(t)}{\sqrt{\mathcal{E}_2}} \tag{2.2–39}$$

In general, the orthogonalization of the kth function leads to

$$\phi_k(t) = \frac{\gamma_k(t)}{\sqrt{\mathcal{E}_k}} \tag{2.2–40}$$

where

$$\gamma_k(t) = s_k(t) - \sum_{i=1}^{k-1} c_{ki}\phi_i(t) \tag{2.2–41}$$

$$c_{ki} = \langle s_k(t), \phi_i(t)\rangle = \int_{-\infty}^{\infty} s_k(t)\phi_i^*(t)\,dt, \qquad i = 1, 2, \ldots, k-1 \tag{2.2–42}$$

$$\mathcal{E}_k = \int_{-\infty}^{\infty} \gamma_k^2(t)\,dt \tag{2.2–43}$$

Thus, the orthogonalization process is continued until all the M signal waveforms $\{s_m(t)\}$ have been exhausted and $N \leq M$ orthonormal waveforms have been constructed. The dimensionality N of the signal space will be equal to M if all the signal waveforms are linearly independent, i.e., none of the signal waveforms is a linear combination of the other signal waveforms.

EXAMPLE 2.2–3. Let us apply the Gram-Schmidt procedure to the set of four waveforms illustrated in Figure 2.2–1. The waveform $s_1(t)$ has energy $\mathcal{E}_1 = 2$, so that

$$\phi_1(t) = \sqrt{\frac{1}{2}}\, s_1(t)$$

Next we observe that $c_{21} = 0$; hence, $s_2(t)$ and $\phi_1(t)$ are orthogonal. Therefore, $\phi_2(t) = s_2(t)/\sqrt{\mathcal{E}_2} = \sqrt{\frac{1}{2}}\, s_2(t)$. To obtain $\phi_3(t)$, we compute c_{31} and c_{32}, which are $c_{31} = \sqrt{2}$ and $c_{23} = 0$. Thus,

$$\gamma_3(t) = s_3(t) - \sqrt{2}\phi_1(t) = \begin{cases} -1 & 2 \leq t \leq 3 \\ 0 & \text{otherwise} \end{cases}$$

Since $\gamma_3(t)$ has unit energy, it follows that $\phi_3(t) = \gamma_3(t)$. Determining $\phi_4(t)$, we find that $c_{41} = -\sqrt{2}$, $c_{42} = 0$, and $c_{43} = 1$. Hence,

$$\gamma_4(t) = s_4(t) + \sqrt{2}\phi_1(t) - \phi_3(t) = 0$$

Consequently, $s_4(t)$ is a linear combination of $\phi_1(t)$ and $\phi_3(t)$ and, hence, $\phi_4(t) = 0$. The three orthonormal functions are illustrated in Figure 2.2–1(b).

Once we have constructed the set of orthonormal waveforms $\{\phi_n(t)\}$, we can express the M signals $\{s_m(t)\}$ as linear combinations of the $\{\phi_n(t)\}$. Thus, we may write

$$s_m(t) = \sum_{n=1}^{N} s_{mn}\phi_n(t), \qquad m = 1, 2, \ldots, M \tag{2.2–44}$$

Based on the expression in Equation 2.2–44, each signal may be represented by the vector

$$\boldsymbol{s}_m = [s_{m1}\, s_{m2} \cdots s_{mN}]^t \tag{2.2–45}$$

or, equivalently, as a point in the N-dimensional (in general, complex) signal space with coordinates $\{s_{mn},\ n = 1, 2, \ldots, N\}$. Therefore, a set of M signals $\{s_m(t)\}_{m=1}^{M}$ can be represented by a set of M vectors $\{\boldsymbol{s}_m\}_{m=1}^{M}$ in the N-dimensional space, where $N \leq M$. The corresponding set of vectors is called the *signal space representation*, or *constellation*, of $\{s_m(t)\}_{m=1}^{M}$. If the original signals are real, then the corresponding vector representations are in $\mathbb{R}^N$; and if the signals are complex, then the vector representations are in $\mathbb{C}^N$. Figure 2.2–2 demonstrates the process of obtaining the vector equivalent from a signal (signal-to-vector mapping) and vice versa (vector-to-signal mapping).

From the orthonormality of the basis $\{\phi_n(t)\}$ it follows that

$$\mathcal{E}_m = \int_{-\infty}^{\infty} |s_m(t)|^2\, dt = \sum_{n=1}^{N} |s_{mn}|^2 = \|\boldsymbol{s}_m\|^2 \tag{2.2–46}$$

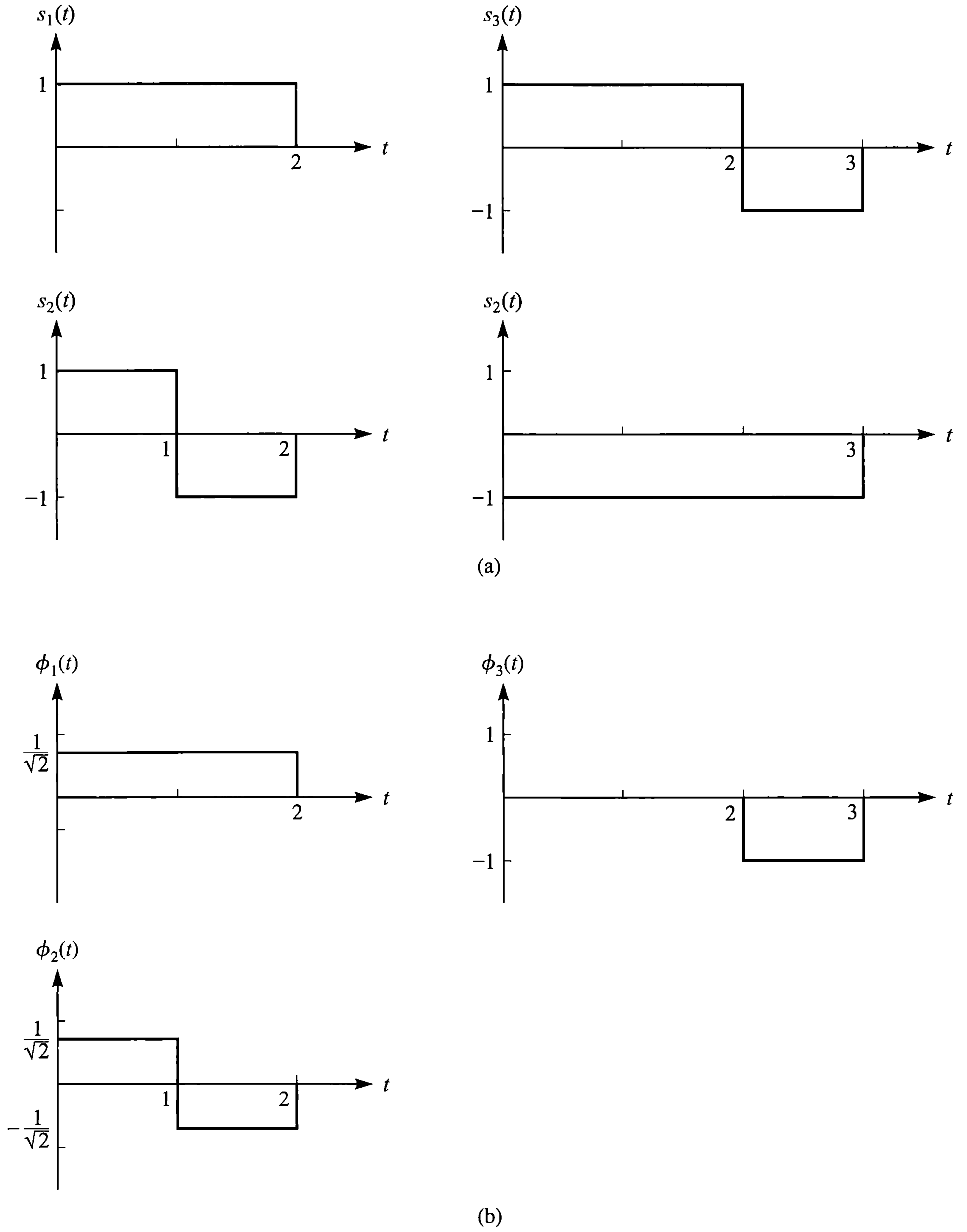

$s_1(t)$
1
2
t
$s_3(t)$
1
2
3
−1
t
$s_2(t)$
1
1
2
−1
t
$s_2(t)$
1
3
−1
t
(a)
$\phi_1(t)$
$\frac{1}{\sqrt{2}}$
2
t
$\phi_3(t)$
1
2
3
−1
t
$\phi_2(t)$
$\frac{1}{\sqrt{2}}$
1
2
$-\frac{1}{\sqrt{2}}$
t
(b)

FIGURE 2.2–1
Gram-Schmidt orthogonalization of the signal $\{s_m(t), m = 1, 2, 3, 4\}$ and the corresponding orthonormal basis.

The energy in the mth signal is simply the square of the length of the vector or, equivalently, the square of the Euclidean distance from the origin to the point $\mathbf{s}_m$ in the N-dimensional space. Thus, any signal can be represented geometrically as a point in the signal space spanned by the orthonormal functions $\{\phi_n(t)\}$. From the orthonormality of the basis it also follows that

$$\langle s_k(t), s_l(t)\rangle = \langle \mathbf{s}_k, \mathbf{s}_l\rangle \tag{2.2–47}$$

This shows that the inner product of two signals is equal to the inner product of the corresponding vectors.

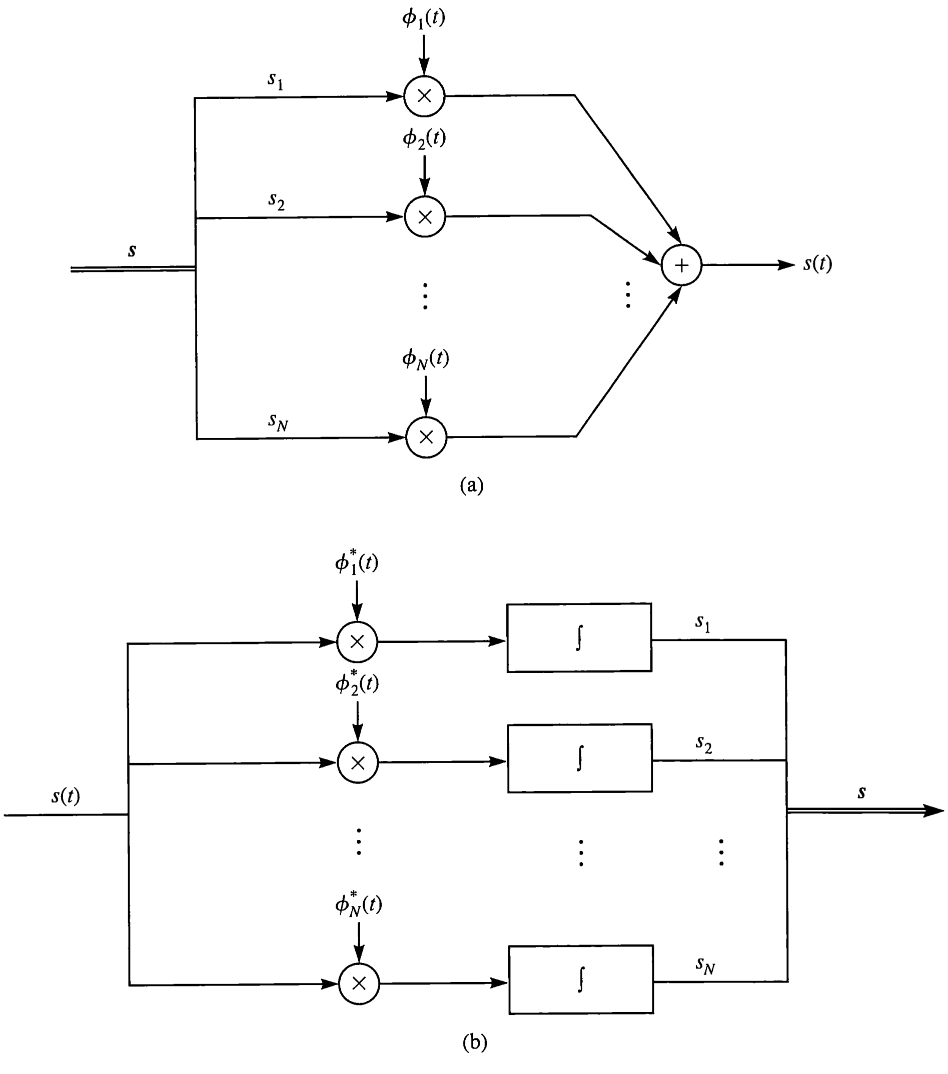


FIGURE 2.2–2
Vector to signal (a), and signal to vector (b) mappings.

EXAMPLE 2.2–4. Let us obtain the vector representation of the four signals shown in Figure 2.2–1(a) by using the orthonormal set of functions in Figure 2.2–1(b). Since the dimensionality of the signal space is $N = 3$, each signal is described by three components. The signal $s_1(t)$ is characterized by the vector $\boldsymbol{s}_1 = (\sqrt{2}, 0, 0)^t$. Similarly, the signals $s_2(t)$, $s_3(t)$, and $s_4(t)$ are characterized by the vectors $\boldsymbol{s}_2 = (0, \sqrt{2}, 0)^t$, $\boldsymbol{s}_3 = (\sqrt{2}, 0, 1)^t$, and $\boldsymbol{s}_4 = (-\sqrt{2}, 0, 1)^t$, respectively. These vectors are shown in Figure 2.2–3. Their lengths are $\|\boldsymbol{s}_1\| = \sqrt{2}$, $\|\boldsymbol{s}_2\| = \sqrt{2}$, $\|\boldsymbol{s}_3\| = \sqrt{3}$, and $\|\boldsymbol{s}_4\| = \sqrt{3}$, and the corresponding signal energies are $\mathcal{E}_k = \|\boldsymbol{s}_k\|^2$, $k = 1, 2, 3, 4$.

We have demonstrated that a set of M finite energy waveforms $\{s_m(t)\}$ can be represented by a weighted linear combination of orthonormal functions $\{\phi_n(t)\}$ of dimensionality $N \leq M$. The functions $\{\phi_n(t)\}$ are obtained by applying the Gram-Schmidt orthogonalization procedure on $\{s_m(t)\}$. It should be emphasized, however, that the functions $\{\phi_n(t)\}$ obtained from the Gram-Schmidt procedure are not unique. If we

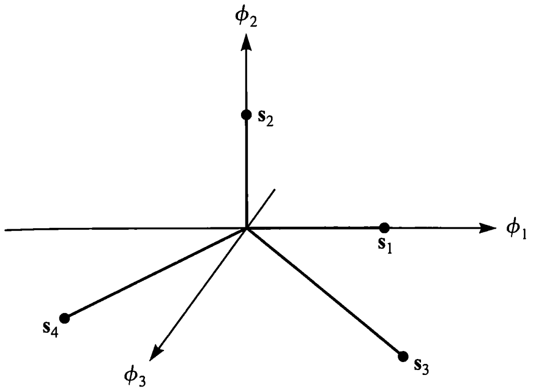


FIGURE 2.2–3
The four signal vectors represented as points in three-dimensional space.

alter the order in which the orthogonalization of the signals $\{s_m(t)\}$ is performed, the orthonormal waveforms will be different and the corresponding vector representation of the signals $\{s_m(t)\}$ will depend on the choice of the orthonormal functions $\{\phi_n(t)\}$. Nevertheless, the dimensionality of the signal space N will not change, and the vectors $\{\boldsymbol{s}_m\}$ will retain their geometric configuration; i.e., their lengths and their inner products will be invariant to the choice of the orthonormal functions $\{\phi_n(t)\}$.

EXAMPLE 2.2–5. An alternative set of orthonormal functions for the four signals in Figure 2.2–1(a) is illustrated in Figure 2.2–4(a). By using these functions to expand $\{s_n(t)\}$, we obtain the corresponding vectors $\boldsymbol{s}_1 = (1, 1, 0)^t$, $\boldsymbol{s}_2 = (1, -1, 0)^t$, $\boldsymbol{s}_3 = (1, 1, -1)^t$, and $\boldsymbol{s}_4 = (-1, -1, -1)^t$, which are shown in Figure 2.2–4(b). Note that the vector lengths are identical to those obtained from the orthonormal functions $\{\phi_n(t)\}$.

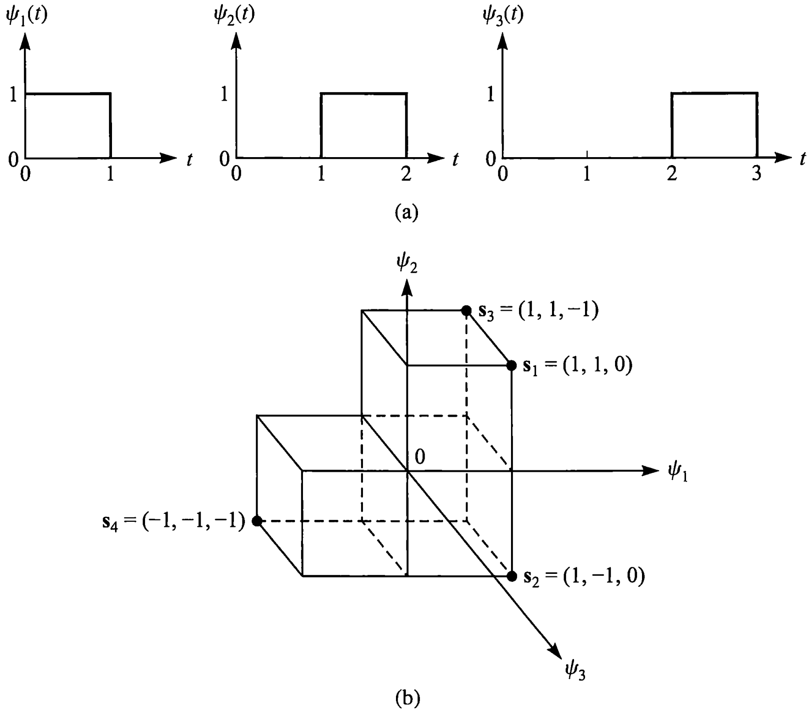


FIGURE 2.2–4
An alternative set of orthonormal functions for the four signals in Figure 2.2–1(a) and the corresponding signal points.

Bandpass and Lowpass Orthonormal Basis

Let us consider the case in which the signal waveforms are bandpass and represented as

$$s_m(t) = \mathrm{Re}\left[s_{ml}(t)e^{j2\pi f_0 t}\right], \qquad m = 1, 2, \ldots, M \tag{2.2–48}$$

where $\{s_{ml}(t)\}$ denotes the lowpass equivalent signals. Recall from Section 2.1–1 that if two lowpass equivalent signals are orthogonal, the corresponding bandpass signals are orthogonal too. Therefore, if $\{\phi_{nl}(t), n = 1, \ldots, N\}$ constitutes an orthonormal basis for the set of lowpass signals $\{s_{ml}(t)\}$, then the set $\{\phi_n(t), n = 1, \ldots, N\}$ where

$$\phi_n(t) = \sqrt{2}\,\mathrm{Re}\left[\phi_{nl}(t)e^{j2\pi f_0 t}\right] \tag{2.2–49}$$

is a set of orthonormal signals, where $\sqrt{2}$ is a normalization factor to make sure each $\phi_n(t)$ has unit energy. However, this set is not necessarily an orthonormal *basis* for expansion of $\{s_m(t), m = 1, \ldots, M\}$. In other words, there is no guarantee that this set is a complete basis for expansion of the set of signals $\{s_m(t), m = 1, \ldots, M\}$. Here our goal is to see how an orthonormal basis for representation of bandpass signals can be obtained from an orthonormal basis used for representation of the lowpass equivalents of the bandpass signals.

Since we have

$$s_{ml}(t) = \sum_{n=1}^{N} s_{mln}\phi_{nl}(t), \qquad m = 1, \ldots, M \tag{2.2–50}$$

where

$$s_{mln} = \langle s_{ml}(t), \phi_{nl}(t)\rangle, \qquad m = 1, \ldots, M, \qquad n = 1, \ldots, N \tag{2.2–51}$$

from Equations 2.2–48 and 2.2–50 we can write

$$s_m(t) = \mathrm{Re}\left[\left(\sum_{n=1}^{N} s_{mln}\phi_{nl}(t)\right) e^{j2\pi f_0 t}\right], \qquad m = 1, \ldots, M \tag{2.2–52}$$

or

$$s_m(t) = \mathrm{Re}\left[\sum_{n=1}^{N} s_{mln}\phi_{nl}(t)\right]\cos 2\pi f_0 t - \mathrm{Im}\left[\sum_{n=1}^{N} s_{mln}\phi_{nl}(t)\right]\sin 2\pi f_0 t \tag{2.2–53}$$

In Problem 2.6 we will see that when an orthonormal set of signals $\{\phi_{nl}(t), n = 1, \ldots, N\}$ constitutes an N-dimensional complex basis for representation of $\{s_{ml}(t), m = 1, \ldots, M\}$, then the set $\{\phi_n(t), \widetilde{\phi}_n(t), n = 1, \ldots, N\}$, where

$$\begin{aligned}\phi_n(t) &= \sqrt{2}\,\mathrm{Re}\left[\phi_{nl}(t)e^{j2\pi f_0 t}\right] = \sqrt{2}\phi_{ni}(t)\cos 2\pi f_0 t - \sqrt{2}\phi_{nq}(t)\sin 2\pi f_0 t\\ \widetilde{\phi}_n(t) &= -\sqrt{2}\,\mathrm{Im}\left[\phi_{nl}(t)e^{j2\pi f_0 t}\right] = -\sqrt{2}\phi_{ni}(t)\sin 2\pi f_0 t - \sqrt{2}\phi_{nq}(t)\cos 2\pi f_0 t\end{aligned} \tag{2.2–54}$$

constitutes a $2N$-dimensional orthonormal basis that is *sufficient* for representation of M bandpass signals

$$s_m(t) = \mathrm{Re}\left[s_{ml}(t)e^{j2\pi f_0 t}\right], \qquad m = 1, \ldots, M \tag{2.2–55}$$

In some cases not all basis functions in the set of basis given by Equation 2.2–54 are necessary, and only a subset of them would be sufficient to expand the bandpass signals. In Problem 2.7 we will further show that

$$\widetilde{\phi}(t) = -\widehat{\phi}(t) \tag{2.2–56}$$

where $\widehat{\phi}(t)$ denotes the Hilbert transform of $\phi(t)$.

From Equation 2.2–52 we have

$$\begin{aligned} s_m(t) &= \mathrm{Re}\left[\left(\sum_{n=1}^{N} s_{mln}\phi_{nl}(t)\right) e^{j2\pi f_0 t}\right] \\ &= \sum_{n=1}^{N} \mathrm{Re}\left[(s_{mln}\phi_{nl}(t))\, e^{j2\pi f_0 t}\right] \\ &= \sum_{n=1}^{N}\left[\frac{s_{mln}^{(r)}}{\sqrt{2}}\phi_n(t) + \frac{s_{mln}^{(i)}}{\sqrt{2}}\widetilde{\phi}_n(t)\right] \end{aligned} \tag{2.2–57}$$

where we have assumed that $s_{mln} = s_{mln}^{(r)} + j s_{mln}^{(i)}$. Equations 2.2–54 and 2.2–57 show how a bandpass signal can be expanded in terms of the basis used for expansion of its lowpass equivalent. In general, lowpass signals can be represented by an N-dimensional complex vector, and the corresponding bandpass signal can be represented by $2N$-dimensional real vectors. If the complex vector

$$\mathbf{s}_{ml} = (s_{ml1}, s_{ml2}, \ldots, s_{mlN})^t$$

is a vector representation for the lowpass signal $s_{ml}(t)$ using the lowpass basis $\{\phi_{nl}(t), n = 1, \ldots, N\}$, then the vector

$$\mathbf{s}_m = \left(\frac{s_{ml1}^{(r)}}{\sqrt{2}}, \frac{s_{ml2}^{(r)}}{\sqrt{2}}, \ldots, \frac{s_{mlN}^{(r)}}{\sqrt{2}}, \frac{s_{ml1}^{(i)}}{\sqrt{2}}, \frac{s_{ml2}^{(i)}}{\sqrt{2}}, \ldots, \frac{s_{mlN}^{(i)}}{\sqrt{2}}\right)^t \tag{2.2–58}$$

is a vector representation of the bandpass signal

$$s_m(t) = \mathrm{Re}\left[s_{ml}(t)e^{j2\pi f_0 t}\right]$$

when the bandpass basis $\{\phi_n(t), \widetilde{\phi}_n(t), n = 1, \ldots, N\}$ given by Equations 2.2–54 and 2.2–57 is used.

EXAMPLE 2.2–6. Let us assume M bandpass signals are defined by

$$s_m(t) = \mathrm{Re}\left[A_m g(t)e^{j2\pi f_0 t}\right] \tag{2.2–59}$$

where A_m's are arbitrary complex numbers and $g(t)$ is a real lowpass signal with energy $\mathcal{E}_g$. The lowpass equivalent signals are given by

$$s_{ml}(t) = A_m g(t)$$

and therefore the unit-energy signal $\phi(t)$ defined by

$$\phi(t) = \frac{g(t)}{\sqrt{\mathcal{E}_g}}$$

is sufficient to expand all $s_{ml}(t)$'s.

We have

$$s_{ml}(t) = A_m \sqrt{\mathcal{E}_g}\, \phi(t)$$

thus, corresponding to each $s_{ml}(t)$ we have a single complex scalar $A_m \sqrt{\mathcal{E}_g} = \left(A_m^{(r)} + jA_m^{(i)}\right) \sqrt{\mathcal{E}_g}$; i.e., the lowpass signals constitute one complex dimension (or, equivalently, two real dimensions). From Equation 2.2–54 we conclude that

$$\phi(t) = \sqrt{\frac{2}{\mathcal{E}_g}}\, g(t) \cos 2\pi f_0 t$$
$$\widetilde{\phi}(t) = -\sqrt{\frac{2}{\mathcal{E}_g}}\, g(t) \sin 2\pi f_0 t$$

can be used as a basis for expansion of the bandpass signals.

Using this basis and Equation 2.2–57, we have

$$\begin{aligned} s_m(t) &= A_m^{(r)} \sqrt{\frac{\mathcal{E}_g}{2}}\, \phi(t) + A_m^{(i)} \sqrt{\frac{\mathcal{E}_g}{2}}\, \widetilde{\phi}(t) \\ &= A_m^{(r)} g(t) \cos 2\pi f_0 t - A_m^{(i)} g(t) \sin 2\pi f_0 t \end{aligned}$$

which agrees with the straightforward expansion of Equation 2.2–59. Note that in the special case where all A_m's are real, $\phi(t)$ is sufficient to represent the bandpass signals and $\widetilde{\phi}(t)$ is not necessary.

2.3
SOME USEFUL RANDOM VARIABLES

In subsequent chapters, we shall encounter several different types of random variables. In this section we list these frequently encountered random variables, their probability density functions (PDFs), their cumulative distribution functions (CDFs), and their moments. Our main emphasis will be on the Gaussian random variable and many random variables that are derived from the Gaussian random variable.

The Bernoulli Random Variable

The Bernoulli random variable is a discrete binary-valued random variable taking values 1 and 0 with probabilities p and $1 - p$, respectively. Therefore the probability mass function (PMF) for this random variable is given by

$$\mathrm{P}[X = 1] = p \qquad \mathrm{P}[X = 0] = 1 - p \tag{2.3–1}$$

The mean and variance of this random variable are given by

$$\begin{aligned} \mathrm{E}[X] &= p \\ \mathrm{VAR}[X] &= p(1 - p) \end{aligned} \tag{2.3–2}$$

The Binomial Random Variable

The binomial random variable models the sum of n independent Bernoulli random variables with common parameter p. The PMF of this random variable is given by

$$\mathrm{P}[X = k] = \binom{n}{k} p^k (1-p)^{n-k}, \quad k = 0, 1, \ldots, n \tag{2.3–3}$$

For this random variable we have

$$\begin{aligned} \mathrm{E}[X] &= np \\ \mathrm{VAR}[X] &= np(1-p) \end{aligned} \tag{2.3–4}$$

This random variable models, for instance, the number of errors when n bits are transmitted over a communication channel and the probability of error for each bit is p.

The Uniform Random Variable

The uniform random variable is a continuous random variable with PDF

$$p(x) = \begin{cases} \frac{1}{b-a} & a \le x \le b \\ 0 & \text{otherwise} \end{cases} \tag{2.3–5}$$

where $b > a$ and the interval $[a, b]$ is the range of the random variable. Here we have

$$\mathrm{E}[X] = \frac{b-a}{2} \tag{2.3–6}$$

$$\mathrm{VAR}[X] = \frac{(b-a)^2}{12} \tag{2.3–7}$$

The Gaussian (Normal) Random Variable

The Gaussian random variable is described in terms of two parameters $m \in \mathbb{R}$ and $\sigma > 0$ by the PDF

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-m)^2}{2\sigma^2}} \tag{2.3–8}$$

We usually use the shorthand form $\mathcal{N}(m, \sigma^2)$ to denote the PDF of Gaussian random variables and write $X \sim \mathcal{N}(m, \sigma^2)$. For this random variable

$$\begin{aligned} \mathrm{E}[X] &= m \\ \mathrm{VAR}[X] &= \sigma^2 \end{aligned} \tag{2.3–9}$$

A Gaussian random variable with $m = 0$ and $\sigma = 1$ is called a *standard normal.* A function closely related to the Gaussian random variable is the Q function defined as

$$Q(x) = \mathrm{P}[\mathcal{N}(0, 1) > x] = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}}\, dt \tag{2.3–10}$$

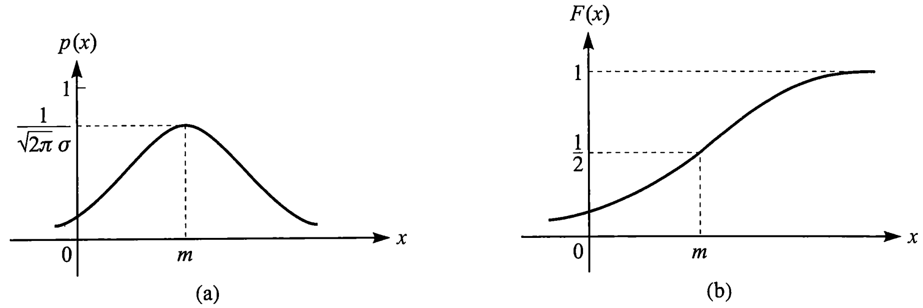


FIGURE 2.3–1
PDF and CDF of a Gaussian random variable.

The CDF of a Gaussian random variable is given by

$$\begin{aligned} F(x) &= \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(t-m)^2}{2\sigma^2}}\, dt \\ &= 1 - \int_{x}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(t-m)^2}{2\sigma^2}}\, dt \\ &= 1 - \int_{\frac{x-m}{\sigma}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}\, du \\ &= 1 - Q\left(\frac{x-m}{\sigma}\right) \end{aligned} \tag{2.3–11}$$

where we have introduced the change of variable $u = (t - m)/\sigma$. The PDF and the CDF of a Gaussian random variable are shown in Figure 2.3–1.

In general if $X \sim \mathcal{N}(m, \sigma^2)$, then

$$\begin{aligned} \mathrm{P}[X > \alpha] &= Q\left(\frac{\alpha - m}{\sigma}\right) \\ \mathrm{P}[X < \alpha] &= Q\left(\frac{m - \alpha}{\sigma}\right) \end{aligned} \tag{2.3–12}$$

Following are some of the important properties of the Q function:

$$Q(0) = \frac{1}{2} \qquad Q(\infty) = 0 \tag{2.3–13}$$

$$Q(-\infty) = 1 \qquad Q(-x) = 1 - Q(x) \tag{2.3–14}$$

Some useful bounds for the Q function for $x > 0$ are

$$\begin{aligned} Q(x) &\le \frac{1}{2} e^{-\frac{x^2}{2}} \\ Q(x) &< \frac{1}{x\sqrt{2\pi}} e^{-\frac{x^2}{2}} \\ Q(x) &> \frac{x}{(1+x^2)\sqrt{2\pi}} e^{-\frac{x^2}{2}} \end{aligned} \tag{2.3–15}$$

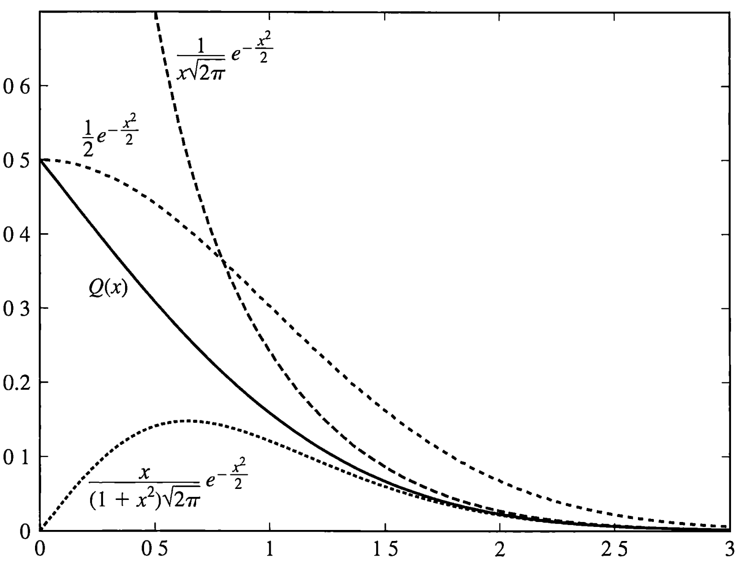


FIGURE 2.3–2
Plot of $Q(x)$ and its upper and lower bounds.

From the last two bounds we conclude that for large x we have

$$Q(x) \approx \frac{1}{x\sqrt{2\pi}}e^{-\frac{x^2}{2}} \tag{2.3–16}$$

A plot of the Q function bounds is given in Figure 2.3–2. Tables 2.3–1 and 2.3–2 give values of the Q function.

TABLE 2.3–1
Table of Q Function Values

x	$Q(x)$	x	$Q(x)$	x	$Q(x)$	x	$Q(x)$
0	0.500000	1.8	0.035930	3.6	0.000159	5.4	3.3320×10^{-8}
0.1	0.460170	1.9	0.028717	3.7	0.000108	5.5	1.8990×10^{-8}
0.2	0.420740	2	0.022750	3.8	7.2348×10^{-5}	5.6	1.0718×10^{-8}
0.3	0.382090	2.1	0.017864	3.9	4.8096×10^{-5}	5.7	5.9904×10^{-9}
0.4	0.344580	2.2	0.013903	4	3.1671×10^{-5}	5.8	3.3157×10^{-9}
0.5	0.308540	2.3	0.010724	4.1	2.0658×10^{-5}	5.9	1.8175×10^{-9}
0.6	0.274250	2.4	0.008198	4.2	1.3346×10^{-5}	6	9.8659×10^{-10}
0.7	0.241960	2.5	0.006210	4.3	8.5399×10^{-6}	6.1	5.3034×10^{-10}
0.8	0.211860	2.6	0.004661	4.4	5.4125×10^{-6}	6.2	2.8232×10^{-10}
0.9	0.184060	2.7	0.003467	4.5	3.3977×10^{-6}	6.3	1.4882×10^{-10}
1	0.158660	2.8	0.002555	4.6	2.1125×10^{-6}	6.4	7.7689×10^{-11}
1.1	0.135670	2.9	0.001866	4.7	1.3008×10^{-6}	6.5	4.0160×10^{-11}
1.2	0.115070	3	0.001350	4.8	7.9333×10^{-7}	6.6	2.0558×10^{-11}
1.3	0.096800	3.1	0.000968	4.9	4.7918×10^{-7}	6.7	1.0421×10^{-11}
1.4	0.080757	3.2	0.000687	5	2.8665×10^{-7}	6.8	5.2309×10^{-12}
1.5	0.066807	3.3	0.000483	5.1	1.6983×10^{-7}	6.9	2.6001×10^{-12}
1.6	0.054799	3.4	0.000337	5.2	9.9644×10^{-8}	7	1.2799×10^{-12}
1.7	0.044565	3.5	0.000233	5.3	5.7901×10^{-8}	7.1	6.2378×10^{-13}

TABLE 2.3–2
Selected Q Function Values

$Q(x)$	x
10^{-1}	1.2816
10^{-2}	2.3263
10^{-3}	3.0902
10^{-4}	3.7190
10^{-5}	4.2649
10^{-6}	4.7534
10^{-7}	5.1993
0.5×10^{-5}	4.4172
0.25×10^{-5}	4.5648
0.667×10^{-5}	4.3545

Another function closely related to the Q function is the *complementary error function*, defined as

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2}\, dt \tag{2.3–17}$$

The complementary error function is related to the Q function as follows:

$$\begin{aligned} Q(x) &= \frac{1}{2}\operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right) \\ \operatorname{erfc}(x) &= 2Q(\sqrt{2}x) \end{aligned} \tag{2.3–18}$$

The characteristic function[†] of a Gaussian random variable is given by

$$\Phi_X(\omega) = e^{j\omega m - \frac{1}{2}\omega^2\sigma^2} \tag{2.3–19}$$

Problem 2.21 shows that for an $\mathcal{N}(m, \sigma^2)$ random variable we have

$$\mathrm{E}\left[(X-m)^n\right] = \begin{cases} 1 \times 3 \times 5 \times \cdots \times (2k-1)\sigma^{2k} = \frac{(2k)!\sigma^{2k}}{2^k k!} & \text{for } n = 2k \\ 0 & \text{for } n = 2k+1 \end{cases} \tag{2.3–20}$$

from which we can obtain moments of the Gaussian random variable.

The sum of n independent Gaussian random variables is a Gaussian random variable whose mean and variance are the sum of the means and the sum of the variances of the random variables, respectively.

[†] Recall that for any random variable X, the *characteristic function* is defined by $\Phi_X(\omega) = E[e^{j\omega X}]$. The *moment generating function* (MGF) is defined by $\Theta_X(t) = E[e^{tX}]$. Obviously, $\Theta(t) = \Phi(-jt)$ and $\Phi(\omega) = \Theta(j\omega)$.

The Chi-Square (χ^2) Random Variable

If $\{X_i, i = 1, \ldots, n\}$ are iid (independent and identically distributed) zero-mean Gaussian random variables with common variance σ^2 and we define

$$X = \sum_{i=1}^{n} X_i^2$$

then X is a χ^2 *random variable with n degrees of freedom.* The PDF of this random variable is given by

$$p(x) = \begin{cases} \frac{1}{2^{n/2}\Gamma(\frac{n}{2})\sigma^n} x^{\frac{n}{2}-1} e^{-\frac{x}{2\sigma^2}} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–21}$$

where $\Gamma(x)$ is the *gamma function* defined by

$$\Gamma(x) = \int_0^{\infty} t^{x-1} e^{-t}\, dt, \tag{2.3–22}$$

The gamma function has simple poles at $x = 0, -1, -2, -3, \ldots$ and satisfies the following properties. The gamma function can be thought of as a generalization of the notion of factorial.

$$\begin{aligned} \Gamma(x+1) &= x\Gamma(x), \\ \Gamma(1) &= 1 \\ \Gamma\left(\frac{1}{2}\right) &= \sqrt{\pi} \\ \Gamma\left(\frac{n}{2}+1\right) &= \begin{cases} \left(\frac{n}{2}\right)! & n \text{ even and positive} \\ \sqrt{\pi}\, \frac{n(n-2)(n-4) \quad .3\times 1}{2^{\frac{n+1}{2}}} & n \text{ odd and positive} \end{cases} \end{aligned} \tag{2.3–23}$$

When n is even, i.e., $n = 2m$, the CDF of the χ^2 random variable with n degrees of freedom has a closed form given by

$$F(x) = \begin{cases} 1 - e^{-\frac{x}{2\sigma^2}} \sum_{k=0}^{m-1} \frac{1}{k!} \left(\frac{x}{2\sigma^2}\right)^k & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–24}$$

The mean and variance of a χ^2 random variable with n degrees of freedom are given by

$$\begin{aligned} \mathrm{E}[X] &= n\sigma^2 \\ \mathrm{VAR}[X] &= 2n\sigma^4 \end{aligned} \tag{2.3–25}$$

The characteristic function for a χ^2 random variable with n degrees of freedom is given by

$$\Phi(\omega) = \left(\frac{1}{1 - 2j\omega\sigma^2}\right)^{\frac{n}{2}} \tag{2.3–26}$$

The special case of a χ^2 random variable with two degrees of freedom is of particular interest. In this case the PDF is given by

$$p(x) = \begin{cases} \frac{1}{2\sigma^2} e^{-\frac{x}{2\sigma^2}} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–27}$$

This is the PDF of an *exponential random variable* with mean equal to $2\sigma^2$.

The χ^2 random variable is a special case of a *gamma random variable.* A gamma random variable is defined by a PDF of the form

$$p(x) = \begin{cases} \frac{\lambda(\lambda x)^{\alpha-1} e^{-\lambda x}}{\Gamma(\alpha)} & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–28}$$

where $\lambda, \alpha > 0$. A χ^2 random variable is a gamma random variable with $\lambda = \frac{1}{2\sigma^2}$ and $\alpha = \frac{n}{2}$.

Plots of the χ^2 random variable with n degrees of freedom for different values of n are shown in Figure 2.3–3.

The Noncentral Chi-Square (χ^2) Random Variable

The *noncentral χ^2 random variable with n degrees of freedom* is defined similarly to a χ^2 random variable in which X_i's are independent Gaussians with common variance σ^2 but with different means denoted by m_i. This random variable has a PDF of the form

$$p(x) = \begin{cases} \frac{1}{2\sigma^2} \left(\frac{x}{s^2}\right)^{\frac{n-2}{4}} e^{-\frac{s^2+x}{2\sigma^2}} I_{\frac{n}{2}-1}\left(\frac{s}{\sigma^2}\sqrt{x}\right) & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–29}$$

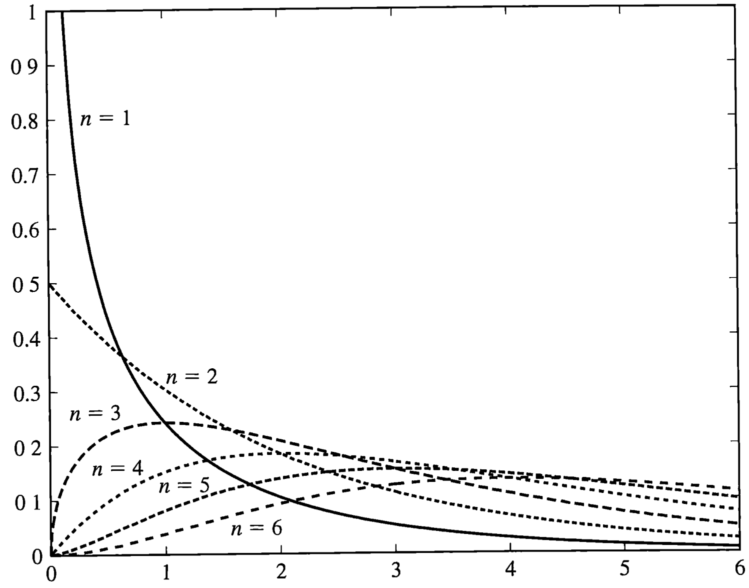


FIGURE 2.3–3
The PDF of the χ^2 random variable for different values of n. All plots are shown for $\sigma = 1$.

where s is defined as

$$s = \sqrt{\sum_{i=1}^{n} m_i^2} \tag{2.3–30}$$

and $I_\alpha(x)$ is the *modified Bessel function of the first kind and order* α given by

$$I_\alpha(x) = \sum_{k=0}^{\infty} \frac{(x/2)^{\alpha+2k}}{k!\,\Gamma(\alpha+k+1)}, \quad x \ge 0 \tag{2.3–31}$$

where $\Gamma(x)$ is the gamma function defined by Equation 2.3–22. The function $I_0(x)$ can be written as

$$I_0(x) = \sum_{k=0}^{\infty} \left(\frac{x^k}{2^k\,k!}\right)^2 \tag{2.3–32}$$

and for $x > 1$ can be approximated by

$$I_0(x) \approx \frac{e^x}{\sqrt{2\pi x}} \tag{2.3–33}$$

Two other expressions for $I_0(x)$, which are used frequently, are

$$\begin{aligned} I_0(x) &= \frac{1}{\pi}\int_0^{\pi} e^{\pm x\cos\phi}\,d\phi \\ I_0(x) &= \frac{1}{2\pi}\int_0^{2\pi} e^{x\cos\phi}\,d\phi \end{aligned} \tag{2.3–34}$$

The CDF of this random variable, when $n = 2m$, can be written in the form

$$F(x) = \begin{cases} 1 - Q_m\left(\frac{s}{\sigma}, \frac{\sqrt{x}}{\sigma}\right) & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–35}$$

where $Q_m(a, b)$ is the *generalized Marcum Q function* and is defined as

$$\begin{aligned} Q_m(a,b) &= \int_b^{\infty} x\left(\frac{x}{a}\right)^{m-1} e^{-(x^2+a^2)/2} I_{m-1}(ax)\,dx \\ &= Q_1(a,b) + e^{-(a^2+b^2)/2}\sum_{k=1}^{m-1}\left(\frac{b}{a}\right)^k I_k(ab) \end{aligned} \tag{2.3–36}$$

In Equation 2.3–36, $Q_1(a, b)$ is the *Marcum Q function* defined as

$$Q_1(a,b) = \int_b^{\infty} x e^{-\frac{a^2+x^2}{2}} I_0(ax)\,dx \tag{2.3–37}$$

or

$$Q_1(a,b) = e^{-\frac{a^2+b^2}{2}}\sum_{k=0}^{\infty}\left(\frac{a}{b}\right)^k I_k(ab), \qquad b \ge a > 0 \tag{2.3–38}$$

This function satisfies the following properties:

$$\begin{aligned} &Q_1(x, 0) = 1 \\ &Q_1(0, x) = e^{-\frac{x^2}{2}} \\ &Q_1(a, b) \approx Q(b-a) \qquad \text{for } b \gg 1 \text{ and } b \gg b-a \end{aligned} \tag{2.3–39}$$

For a noncentral χ^2 random variable, the mean and variance are given by

$$\begin{aligned} \mathrm{E}[X] &= n\sigma^2 + s^2 \\ \mathrm{VAR}[X] &= 2n\sigma^4 + 4\sigma^2 s^2 \end{aligned} \tag{2.3–40}$$

and the characteristic function is given by

$$\Phi(\omega) = \left(\frac{1}{1 - 2j\omega\sigma^2}\right)^{\frac{n}{2}} e^{\frac{j\omega s^2}{1-2j\omega\sigma^2}} \tag{2.3–41}$$

The Rayleigh Random Variable

If X_1 and X_2 are two iid Gaussian random variables each distributed according to $\mathcal{N}(0, \sigma^2)$, then

$$X = \sqrt{X_1^2 + X_2^2} \tag{2.3–42}$$

is a *Rayleigh random variable*. From our discussion of the χ^2 random variables, it is readily seen that a Rayleigh random variable is the square root of a χ^2 random variable with two degrees of freedom. We can also conclude that the Rayleigh random variable is the square root of an exponential random variable as given by Equation 2.3–27. The PDF of a Rayleigh random variable is given by

$$p(x) = \begin{cases} \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–43}$$

and its mean and variance are

$$\begin{aligned} \mathrm{E}[X] &= \sigma\sqrt{\frac{\pi}{2}} \\ \mathrm{VAR}[X] &= \left(2 - \frac{\pi}{2}\right)\sigma^2 \end{aligned} \tag{2.3–44}$$

In general, the nth moment of a Rayleigh random variable is given by

$$\mathrm{E}\left[X^k\right] = (2\sigma^2)^{k/2}\Gamma\left(\frac{k}{2} + 1\right) \tag{2.3–45}$$

and its characteristic function is given by

$$\Phi_X(\omega) = {}_1F_1\left(1, \frac{1}{2}; -\frac{1}{2}\omega^2\sigma^2\right) + j\sqrt{\frac{\pi}{2}}\omega\sigma e^{-\frac{\omega^2\sigma^2}{2}} \tag{2.3–46}$$

where ${}_1F_1\,(a, b; x)$ is the *confluent hypergeometric function* defined by

$$ {}_1F_1\,(a, b; x) = \sum_{k=0}^{\infty} \frac{\Gamma(a+k)\Gamma(b)x^k}{\Gamma(a)\Gamma(b+k)k!}, \qquad b \neq 0, -1, -2, \ldots \tag{2.3–47}$$

The function ${}_1F_1\,(a, b; x)$ can also be written as the integral

$$ {}_1F_1\,(a, b; x) = \frac{\Gamma(b)}{\Gamma(b-a)\Gamma(a)} \int_0^1 e^{xt} t^{a-1} (1-t)^{b-a-1}\, dt \tag{2.3–48}$$

In Beaulieu (1990), it is shown that

$$ {}_1F_1\left(1, \frac{1}{2}; -x\right) = -e^{-x} \sum_{k=0}^{\infty} \frac{x^k}{(2k-1)k!} \tag{2.3–49}$$

The CDF of a Rayleigh random variable can be easily found by integrating the PDF. The result is

$$ F(x) = \begin{cases} 1 - e^{-\frac{x^2}{2\sigma^2}} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–50}$$

The PDF of a Rayleigh random variable is plotted in Figure 2.3–4.

A generalized version of the Rayleigh random variable is obtained when we have n iid zero-mean Gaussian random variables $\{X_i, 1 \leq i \leq n\}$ where each X_i has an $\mathcal{N}(0, \sigma^2)$ distribution. In this case

$$ X = \sqrt{\sum_{i=1}^{n} X_i^2} \tag{2.3–51}$$

has a *generalized Rayleigh distribution*. The PDF for this random variable is given by

$$ p(x) = \begin{cases} \frac{x^{n-1}}{2^{\frac{n-2}{2}} \sigma^n \Gamma(\frac{n}{2})} e^{-\frac{x^2}{2\sigma^2}} & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–52}$$

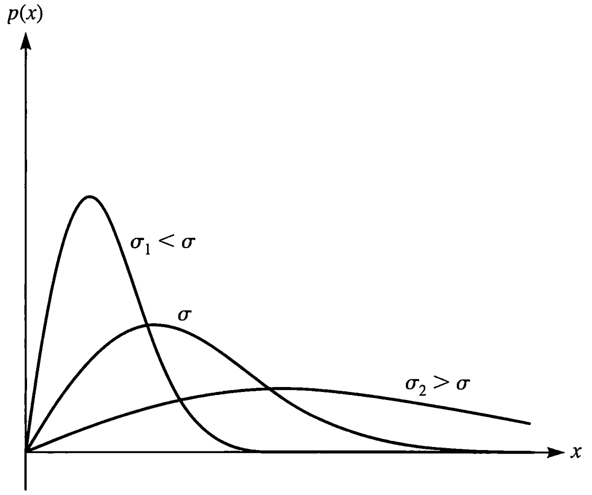


FIGURE 2.3–4
The PDF of the Rayleigh random variable for three different values of σ.

For the generalized Rayleigh, and with $n = 2m$, the CDF is given by

$$F(x) = \begin{cases} 1 - e^{-\frac{x^2}{2\sigma^2}} \sum_{k=0}^{m-1} \frac{1}{k!} \left(\frac{x^2}{2\sigma^2}\right)^k & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–53}$$

The kth moment of a generalized Rayleigh for any integer value of n (even or odd) is given by

$$\mathrm{E}\left[X^k\right] = (2\sigma^2)^{\frac{k}{2}} \frac{\Gamma\left(\frac{n+k}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \tag{2.3–54}$$

The Ricean Random Variable

If X_1 and X_2 are two independent Gaussian random variables distributed according to $\mathcal{N}(m_1, \sigma^2)$ and $\mathcal{N}(m_2, \sigma^2)$ (i.e., the variances are equal and the means may be different), then

$$X = \sqrt{X_1^2 + X_2^2} \tag{2.3–55}$$

is a Ricean random variable with PDF

$$p(x) = \begin{cases} \frac{x}{\sigma^2} I_0\left(\frac{sx}{\sigma^2}\right) e^{-\frac{x^2+s^2}{2\sigma^2}} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–56}$$

where $s = \sqrt{m_1^2 + m_2^2}$ and $I_0(x)$ is given by Equation 2.3–32. It is clear that a Ricean random variable is the square root of a noncentral χ^2 random variable with two degrees of freedom.

It is readily seen that for $s = 0$, the Ricean random variable reduces to a Rayleigh random variable. For large s the Ricean random variable can be well approximated by a Gaussian random variable.

The CDF of a Ricean random variable can be expressed as

$$F(x) = \begin{cases} 1 - Q_1\left(\frac{s}{\sigma}, \frac{x}{\sigma}\right) & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–57}$$

where $Q_1(a, b)$ is defined by Equations 2.3–37 and 2.3–38.

The first two moments of the Ricean random variable are given by

$$\begin{aligned} \mathrm{E}[X] &= \sigma\sqrt{\frac{\pi}{2}}\, {}_1F_1\left(-\frac{1}{2}, 1, -\frac{s^2}{2\sigma^2}\right) \\ &= \sigma\sqrt{\frac{\pi}{2}}\, e^{-\frac{K}{2}} \left[(1+K) I_0\left(\frac{K}{2}\right) + K I_1\left(\frac{K}{2}\right)\right] \\ \mathrm{E}\left[X^2\right] &= 2\sigma^2 + s^2 \end{aligned} \tag{2.3–58}$$

where K is the Rice factor defined in Equation 2.3–60.

In general, the kth moment of this random variable is given by

$$\mathrm{E}\left[X^k\right] = (2\sigma^2)^{\frac{k}{2}}\Gamma\left(1+\frac{k}{2}\right)\,{}_1F_1\left(-\frac{k}{2}, 1; -\frac{s^2}{2\sigma^2}\right) \tag{2.3–59}$$

Another form of the Ricean density function is obtained by defining the *Rice factor* K as

$$K = \frac{s^2}{2\sigma^2} \tag{2.3–60}$$

If we define $A = s^2 + 2\sigma^2$, the Ricean PDF can be written as

$$p(x) = \begin{cases} \frac{2(K+1)}{A} x e^{-\frac{K+1}{A}\left(x^2+\frac{AK}{K+1}\right)} I_0\left(2x\sqrt{\frac{K(K+1)}{A}}\right) & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–61}$$

For the normalized case when $A = 1$ (or, equivalently, when $\mathrm{E}\left[X^2\right] = s^2 + 2\sigma^2 = 1$) this reduces to

$$p(x) = \begin{cases} 2(K+1) x e^{-(K+1)\left(x^2+\frac{K}{K+1}\right)} I_0\left(2x\sqrt{K(K+1)}\right) & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–62}$$

A plot of the PDF of a Ricean random variable for different values of K is shown in Figure 2.3–5.

Similar to the Rayleigh random variable, a *generalized Ricean random variable* can be defined as

$$X = \sqrt{\sum_{i=1}^{n} X_i^2} \tag{2.3–63}$$

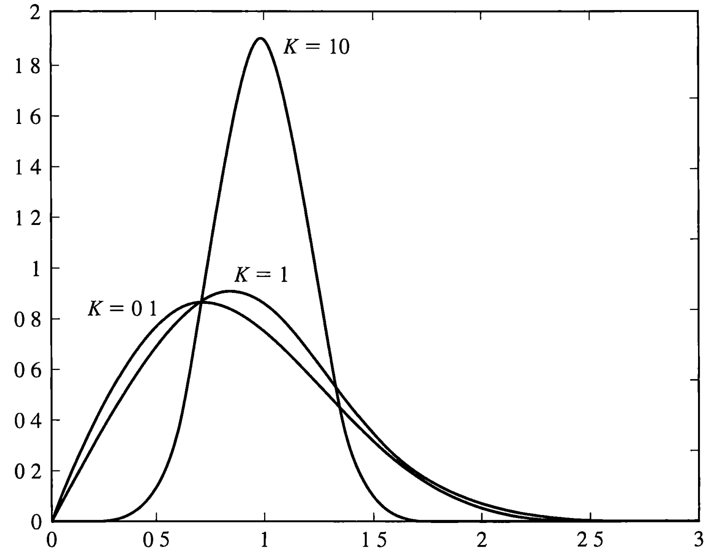


FIGURE 2.3–5
The Ricean PDF for different values of K. For small K this random variable reduces to a Rayleigh random variable, and for large K it is well approximated by a Gaussian random variable.

where X_i's are independent Gaussians with mean m_i and common variance σ^2. In this case the PDF is given by

$$p(x) = \begin{cases} \frac{x^{\frac{n}{2}}}{\sigma^2 s^{\frac{n-2}{2}}} e^{-\frac{x^2+s^2}{2\sigma^2}} I_{\frac{n}{2}-1}\left(\frac{xs}{\sigma^2}\right) & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–64}$$

and the CDF is given by

$$F(x) = \begin{cases} 1 - Q_m\left(\frac{s}{\sigma}, \frac{x}{\sigma}\right) & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–65}$$

where

$$s = \sqrt{\sum_{i=1}^{n} m_i^2}$$

The kth moment of a generalized Ricean is given by

$$\mathrm{E}\left[X^k\right] = (2\sigma^2)^{\frac{k}{2}} e^{-\frac{s^2}{2\sigma^2}} \frac{\Gamma\left(\frac{n+k}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} {}_1F_1\left(\frac{n+k}{2}, \frac{n}{2}; \frac{s^2}{2\sigma^2}\right) \tag{2.3–66}$$

The Nakagami Random Variable

Both the Rayleigh distribution and the Rice distribution are frequently used to describe the statistical fluctuations of signals received from a multipath fading channel. These channel models are considered in Chapters 13 and 14. Another distribution that is frequently used to characterize the statistics of signals transmitted through multipath fading channels is the Nakagami m distribution. The PDF for this distribution is given by Nakagami (1960) as

$$p(x) = \begin{cases} \frac{2}{\Gamma(m)} \left(\frac{m}{\Omega}\right)^m x^{2m-1} e^{-mx^2/\Omega} & x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–67}$$

where Ω is defined as

$$\Omega = \mathrm{E}\left[X^2\right] \tag{2.3–68}$$

and the parameter m is defined as the ratio of moments, called the *fading figure*,

$$m = \frac{\Omega^2}{\mathrm{E}\left[\left(X^2 - \Omega\right)^2\right]}, \qquad m \geq \frac{1}{2} \tag{2.3–69}$$

A normalized version of Equation 2.3–67 may be obtained by defining another random variable $Y = X/\sqrt{\Omega}$ (see Problem 2.42). The nth moment of X is

$$\mathrm{E}\left[X^n\right] = \frac{\Gamma\left(m + \frac{n}{2}\right)}{\Gamma(m)} \left(\frac{\Omega}{m}\right)^{n/2} \tag{2.3–70}$$

The mean and the variance for this random variable are given by

$$
\begin{aligned}
\mathrm{E}[X] &= \frac{\Gamma\left(m+\frac{1}{2}\right)}{\Gamma(m)}\left(\frac{\Omega}{m}\right)^{1/2} \\
\mathrm{VAR}[X] &= \Omega\left(1-\frac{1}{m}\left(\frac{\Gamma\left(m+\frac{1}{2}\right)}{\Gamma(m)}\right)^{2}\right)
\end{aligned}
\tag{2.3–71}
$$

By setting $m = 1$, we observe that Equation 2.3–67 reduces to a Rayleigh PDF. For values of m in the range $\frac{1}{2} \leq m \leq 1$, we obtain PDFs that have larger tails than a Rayleigh-distributed random variable. For values of $m > 1$, the tail of the PDF decays faster than that of the Rayleigh. Figure 2.3–6 illustrates the Nakagami PDF for different values of m.

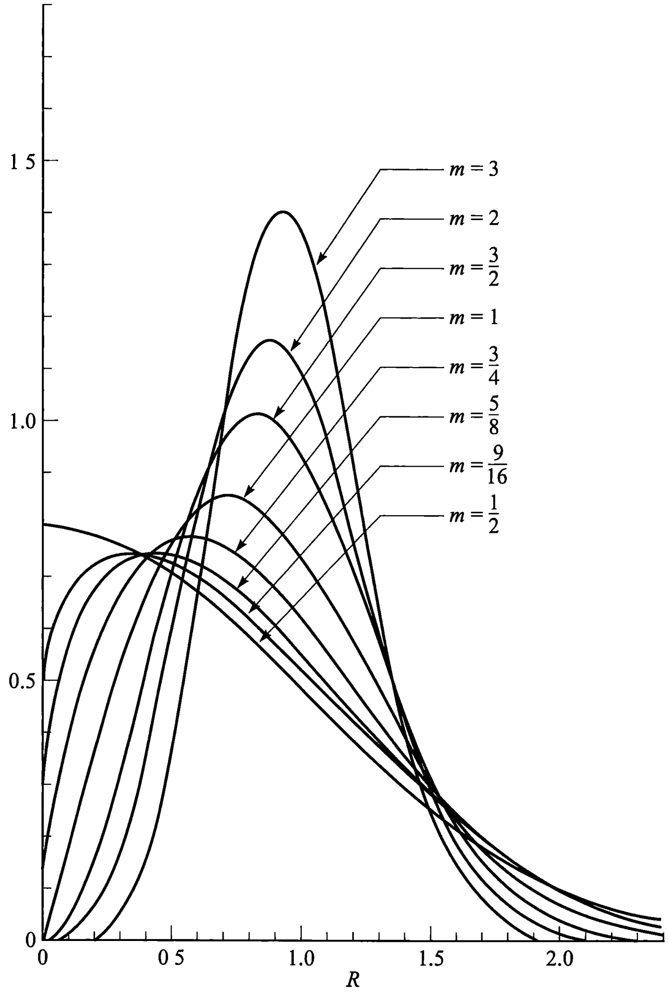


FIGURE 2.3–6
The PDF for the Nakagami m distribution, shown with $\Omega = 1$. m is the fading figure.

The Lognormal Random Variable

Suppose that a random variable Y is normally distributed with mean m and variance σ^2. Let us define a new random variable X that is related to Y through the transformation $Y = \ln X$ (or $X = e^Y$). Then the PDF of X is

$$p(x) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma^2}\,x} e^{-(\ln x - m)^2/2\sigma^2} & x \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.3–72}$$

For this random variable

$$\begin{aligned} \mathrm{E}[X] &= e^{m+\frac{\sigma^2}{2}} \\ \mathrm{VAR}[X] &= e^{2m+\sigma^2}\left(e^{\sigma^2} - 1\right) \end{aligned} \tag{2.3–73}$$

The lognormal distribution is suitable for modeling the effect of shadowing of the signal due to large obstructions, such as tall buildings, in mobile radio communications. Examples of the lognormal PDF are shown in Figure 2.3–7.

Jointly Gaussian Random Variables

An $n \times 1$ column random vector $\boldsymbol{X}$ with components $\{X_i, 1 \leq i \leq n\}$ is called a *Gaussian vector*, and its components are called *jointly Gaussian random variables* or

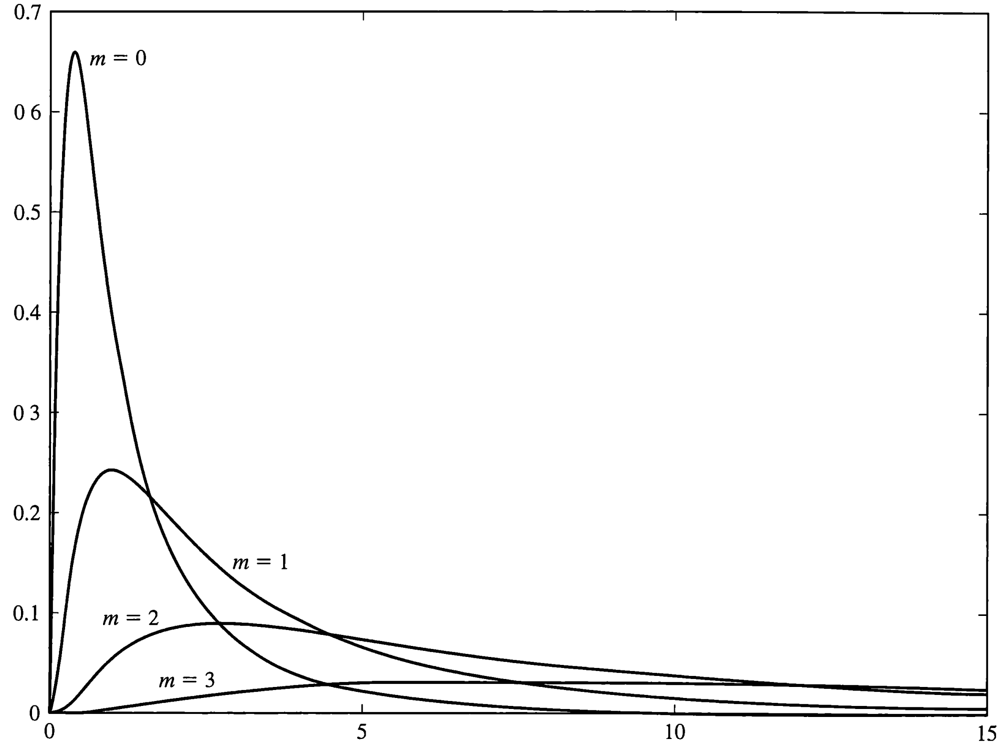


FIGURE 2.3–7
Lognormal PDF with $\sigma = 1$ for different values of m.

multivariate Gaussian random variables if the joint PDF of X_i's can be written as

$$p(\boldsymbol{x}) = \frac{1}{(2\pi)^{n/2}(\det \boldsymbol{C})^{1/2}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{m})^t \boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{m})} \tag{2.3–74}$$

where $\boldsymbol{m}$ and $\boldsymbol{C}$ are the mean vector and covariance matrix, respectively, of $\boldsymbol{X}$ and are given by

$$\begin{aligned} \boldsymbol{m} &= \mathrm{E}[\boldsymbol{X}] \\ \boldsymbol{C} &= \mathrm{E}\left[(\boldsymbol{X}-\boldsymbol{m})(\boldsymbol{X}-\boldsymbol{m})^t\right] \end{aligned} \tag{2.3–75}$$

From this definition it is clear that

$$C_{ij} = \mathrm{COV}\left[X_i, X_j\right] \tag{2.3–76}$$

and therefore $\boldsymbol{C}$ is a symmetric matrix. From elementary probability it is also well known that $\boldsymbol{C}$ is nonnegative definite.

In the special case of $n = 2$, we have

$$\begin{aligned} \boldsymbol{m} &= \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \\ \boldsymbol{C} &= \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \end{aligned} \tag{2.3–77}$$

where

$$\rho = \frac{\mathrm{COV}[X_1, X_2]}{\sigma_1\sigma_2}$$

is the correlation coefficient of the two random variables. In this case the PDF reduces to

$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} e^{-\frac{\left(\frac{x_1-m_1}{\sigma_1}\right)^2 + \left(\frac{x_2-m_2}{\sigma_2}\right)^2 - 2\rho\left(\frac{x_1-m_1}{\sigma_1}\right)\left(\frac{x_2-m_2}{\sigma_2}\right)}{2(1-\rho^2)}} \tag{2.3–78}$$

where m_1, m_2, σ_1^2 and, σ_2^2 are means and variances of the two random variables and ρ is their correlation coefficient. Note that in the special case when $\rho = 0$ (i.e., when the two random variables are uncorrelated), we have

$$p(x_1, x_2) = \mathcal{N}\left(m_1, \sigma_1^2\right) \times \mathcal{N}\left(m_2, \sigma_2^2\right)$$

This means that the two random variables are independent, and therefore for this case independence and uncorrelatedness are equivalent. This property is true for general jointly Gaussian random variables.

Another important property of jointly Gaussian random variables is that linear combinations of jointly Gaussian random variables are also jointly Gaussian. In other words, if $\boldsymbol{X}$ is a Gaussian vector, the random vector $\boldsymbol{Y} = \boldsymbol{AX}$, where the invertible matrix $\boldsymbol{A}$ represents a linear transformation, is also a Gaussian vector whose mean and covariance matrix are given by

$$\begin{aligned} \boldsymbol{m}_Y &= \boldsymbol{A}\boldsymbol{m}_X \\ \boldsymbol{C}_Y &= \boldsymbol{A}\boldsymbol{C}_X\boldsymbol{A}^t \end{aligned} \tag{2.3–79}$$

This property is developed in Problem 2.23.

In summary, jointly Gaussian random variables have the following important properties:

1. For jointly Gaussian random variables, *uncorrelated* is equivalent to *independent*.
2. Linear combinations of jointly Gaussian random variables are themselves jointly Gaussian.
3. The random variables in any subset of jointly Gaussian random variables are jointly Gaussian, and any subset of random variables conditioned on random variables in any other subset is also jointly Gaussian (all joint subsets and all conditional subsets are Gaussian).

We also emphasize that any set of independent Gaussian random variables is jointly Gaussian, but this is not necessarily true for a set of dependent Gaussian random variables.

Table 2.3–3 summarizes some of the properties of the most important random variables.

2.4
BOUNDS ON TAIL PROBABILITIES

Performance analysis of communication systems requires computation of error probabilities of these systems. In many cases, as we will observe in the following chapters, the error probability of a communication system is expressed in terms of the probability that a random variable exceeds a certain value, i.e., in the form of $\mathrm{P}[X > \alpha]$. Unfortunately, in many cases these probabilities cannot be expressed in closed form. In such cases we are interested in finding upper bounds on these tail probabilities. These upper bounds are of the form $\mathrm{P}[X > \alpha] \leq \beta$. In this section we describe different methods for providing and tightening such bounds.

The Markov Inequality

The Markov inequality gives an upper bound on the tail probability of nonnegative random variables. Let us assume that X is a nonnegative random variable, i.e., $p(x) = 0$ for all $x < 0$, and assume $\alpha > 0$ is an arbitrary positive real number. The Markov inequality states that

$$\mathrm{P}[X \geq \alpha] \leq \frac{\mathrm{E}[X]}{\alpha} \tag{2.4–1}$$

TABLE 2.3–3

Properties of Important Random Variables

RV (Parameters)	PDF or PMF	E[X]	VAR[X]	$\Phi_X(\omega) = E\left[e^{j\omega X}\right]$
Bernoulli (p)	$P(X=1) = 1 - P(X=0) = p$ $0 \le p \le 1$	p	$p(1-p)$	$pe^{j\omega} + (1-p)$
Binomial (n, p)	$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$ $0 \le k \le n,\ 0 \le p \le 1$	np	$np(1-p)$	$\left(pe^{j\omega} + (1-p)\right)^n$
Uniform (a, b)	$\frac{1}{b-a},\ a \le x \le b$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$	$\frac{e^{j\omega b} - e^{j\omega a}}{j\omega(b-a)}$
Exponential (λ)	$\lambda e^{-\lambda x},\ \lambda > 0,\ x \ge 0$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$	$\frac{\lambda}{\lambda - j\omega}$
Gaussian (m, σ^2)	$\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-m)^2}{2\sigma^2}}$ $\sigma > 0$	m	σ^2	$e^{j\omega m - \frac{\omega^2\sigma^2}{2}}$
Gamma (λ, α)	$\frac{\lambda(\lambda x)^{\alpha-1} e^{-\lambda x}}{\Gamma(\alpha)}$ $x \ge 0,\ \lambda, \alpha > 0$	$\frac{\alpha}{\lambda}$	$\frac{\alpha}{\lambda^2}$	$\left(\frac{\lambda}{\lambda - j\omega}\right)^\alpha$
$\chi^2(n, \sigma^2)$	$\frac{1}{2^{n/2}\Gamma\left(\frac{n}{2}\right)\sigma^n} x^{\frac{n}{2}-1} e^{-\frac{x}{2\sigma^2}}$ $x, \sigma > 0,\quad n \in \mathbb{N}$	$n\sigma^2$	$2n\sigma^4$	$\left(\frac{1}{1-2j\omega\sigma^2}\right)^{n/2}$
Noncentral $\chi^2(n, s, \sigma^2)$	$\frac{1}{2\sigma^2}\left(\frac{x}{s^2}\right)^{\frac{n-2}{4}} e^{-\frac{s^2+x}{2\sigma^2}} I_{\frac{n}{2}-1}\left(\frac{s}{\sigma^2}\sqrt{x}\right)$ $x, s, \sigma > 0,\quad n \in \mathbb{N}$	$n\sigma^2 + s^2$	$2n\sigma^4 + 4\sigma^2 s^2$	$\left(\frac{1}{1-2j\omega\sigma^2}\right)^{n/2} e^{\frac{j\omega s^2}{1-2jw\sigma^2}}$
Rayleigh (σ^2)	$\frac{x}{\sigma^2} e^{-x^2/2\sigma^2}$ $x, \sigma > 0$	$\sigma\sqrt{\frac{\pi}{2}}$	$\left(2 - \frac{\pi}{2}\right)\sigma^2$	${}_1F_1\left(1, \frac{1}{2}; -\frac{\omega^2\sigma^2}{2}\right) + j\sqrt{\frac{\pi}{2}}\omega\sigma e^{-\frac{\omega^2\sigma^2}{2}}$
Ricean (σ^2, s)	$\frac{x}{\sigma^2} I_0\left(\frac{xs}{\sigma^2}\right) e^{-\frac{x^2+s^2}{2\sigma^2}}$ $x, s, \sigma > 0$	$\sigma\sqrt{\frac{\pi}{2}}\ {}_1F_1\left(-\frac{1}{2}, 1, -\frac{s^2}{2\sigma^2}\right)$	$2\sigma^2 + s^2 - (\mathrm{E}[X])^2$	—
Jointly Gaussian ($\boldsymbol{m}, \boldsymbol{C}$)	$\frac{1}{(2\pi)^{n/2}\det(\boldsymbol{C})} e^{-\frac{1}{2}\left[(\boldsymbol{x}-\boldsymbol{m})^t \boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{m})\right]}$ $\boldsymbol{C}$ symmetric and positive definite	$\boldsymbol{m}$	$\boldsymbol{C}$ (cov. matrix)	$e^{j\boldsymbol{m}^t\boldsymbol{\omega} - \frac{1}{2}\boldsymbol{\omega}^t\boldsymbol{C}\boldsymbol{\omega}}$

To see this, we observe that

$$\begin{aligned} \mathrm{E}[X] &= \int_0^\infty xp(x)\,dx \\ &\geq \int_\alpha^\infty xp(x)\,dx \\ &\geq \alpha \int_\alpha^\infty xp(x)\,dx \\ &= \alpha\,\mathrm{P}[X \geq \alpha] \end{aligned} \tag{2.4–2}$$

Dividing both sides by α gives the desired inequality.

Chernov Bound

The Chernov bound is a very tight and useful bound that is obtained from the Markov inequality. Unlike the Markov inequality that is applicable only to nonnegative random variables, the Chernov bound can be applied to all random variables.

Let X be an arbitrary random variable, and let δ and ν be arbitrary real numbers ($\nu \neq 0$). Define random variable Y by $Y = e^{\nu X}$ and constant α by $\alpha = e^{\nu\delta}$. Obviously, Y is a nonnegative random variable and α is a positive real number. Applying the Markov inequality to Y and α yields

$$\mathrm{P}\left[e^{\nu X} \geq e^{\nu\delta}\right] \leq \frac{\mathrm{E}\left[e^{\nu X}\right]}{e^{\nu\delta}} = \mathrm{E}\left[e^{\nu(X-\delta)}\right] \tag{2.4–3}$$

The event $\{e^{\nu X} \geq e^{\nu\delta}\}$ is equivalent to the event $\{\nu X \geq \nu\delta\}$ which for positive or negative values of ν is equivalent to $\{X \geq \delta\}$ or $\{X \leq \delta\}$, respectively. Therefore we have

$$\mathrm{P}[X \geq \delta] \leq \mathrm{E}\left[e^{\nu(X-\delta)}\right], \quad \text{for all } \nu > 0 \tag{2.4–4}$$

$$\mathrm{P}[X \leq \delta] \leq \mathrm{E}\left[e^{\nu(X-\delta)}\right], \quad \text{for all } \nu < 0 \tag{2.4–5}$$

Since the two inequalities are valid for all positive and negative values of ν, respectively, it makes sense to find the values of ν that give the tightest possible bounds. To this end, we differentiate the right hand of the inequalities with respect to ν and find its root; this is the value of ν that gives the tightest bound. From this point on, we will consider only the first inequality. The extension to the second inequality is straightforward.

Let us define function $g(\nu)$ to denote the right side of the inequalities, i.e.,

$$g(\nu) = \mathrm{E}\left[e^{\nu(X-\delta)}\right]$$

Differentiating $g(\nu)$, we have

$$g'(\nu) = \mathrm{E}\left[(X-\delta)e^{\nu(X-\delta)}\right] \tag{2.4–6}$$

The second derivative of $g(\nu)$ is given by

$$g''(\nu) = \mathrm{E}\left[(X-\delta)^2\, e^{\nu(X-\delta)}\right]$$

It is easily seen that for all ν, we have $g''(\nu) > 0$ and hence $g(\nu)$ is convex and $g'(\nu)$ is an increasing function, and therefore can have only one root. In addition, since $g(\nu)$ is convex, this single root minimizes $g(\nu)$ and therefore results in the the tightest bound. Putting $g'(\nu) = 0$, we find the root to be obtained by solving the equation

$$\mathrm{E}\left[Xe^{\nu X}\right] = \delta\,\mathrm{E}\left[e^{\nu X}\right] \tag{2.4–7}$$

Equation 2.4–7 has a single root ν^* that gives the tightest bound. The only thing that remains to be checked is to see whether this ν^* satisfies the $\nu^* > 0$ condition. Since $g'(\nu)$ is an increasing function, its only root is positive if $g'(0) < 0$. From Equation 2.4–6 we have

$$g'(0) = \mathrm{E}[X] - \delta$$

therefore $\nu^* > 0$ if and only if $\delta > \mathrm{E}[X]$.

Summarizing, from Equations 2.4–4 and 2.4–5 we conclude

$$\mathrm{P}[X \geq \delta] \leq e^{-\nu^*\delta}\,\mathrm{E}\left[e^{\nu^* X}\right], \quad \text{for } \delta > \mathrm{E}[X] \tag{2.4–8}$$

$$\mathrm{P}[X \leq \delta] \leq e^{-\nu^*\delta}\,\mathrm{E}\left[e^{\nu^* X}\right], \quad \text{for } \delta < \mathrm{E}[X] \tag{2.4–9}$$

where ν^* is the solution of Equation 2.4–7. Equations 2.4–8 and 2.4–9 are known as Chernov bounds. Finding optimal ν^* by solving Equation 2.4–7 is sometimes difficult. In such cases a numerical approximation or an educated guess gives a suboptimal bound. The Chernov bound can also be given in terms of the moment generating function (MGF) $\Theta_X(\nu) = \mathrm{E}\left[e^{\nu X}\right]$ as

$$\mathrm{P}[X \geq \delta] \leq e^{-\nu^*\delta}\Theta_X(\nu^*), \quad \text{for } \delta > \mathrm{E}[X] \tag{2.4–10}$$

$$\mathrm{P}[X \leq \delta] \leq e^{-\nu^*\delta}\Theta_X(\nu^*), \quad \text{for } \delta < \mathrm{E}[X] \tag{2.4–11}$$

EXAMPLE 2.4–1. Consider the Laplace PDF given by

$$p(x) = \frac{1}{2}e^{-|x|} \tag{2.4–12}$$

Let us evaluate the upper tail probability $\mathrm{P}[X \geq \delta]$ for some $\delta > 0$ from the Chernov bound and compare it with the true tail probability, which is

$$\mathrm{P}[X \geq \delta] = \int_{\delta}^{\infty} \frac{1}{2}e^{-x}\,dx = \frac{1}{2}e^{-\delta} \tag{2.4–13}$$

First note that $\mathrm{E}[X] = 0$, and therefore the condition $\delta > \mathrm{E}[X]$ needed to use the upper tail probability in the Chernov bound is satisfied. To solve Equation 2.4–7 for ν^*, we must determine $\mathrm{E}\left[Xe^{\nu X}\right]$ and $\mathrm{E}\left[e^{\nu X}\right]$. For the PDF in Equation 2.4–12, we find that $\mathrm{E}\left[Xe^{\nu X}\right]$ and $\mathrm{E}\left[e^{\nu X}\right]$ converge only if $-1 < \nu < 1$, and for this range of values of ν we have

$$\begin{aligned} \mathrm{E}\left[Xe^{\nu X}\right] &= \frac{2\nu}{(\nu+1)^2(\nu-1)^2} \\ \mathrm{E}\left[e^{\nu X}\right] &= \frac{1}{(1+\nu)(1-\nu)} \end{aligned} \tag{2.4–14}$$

Substituting these values into Equation 2.4–7, we obtain the quadratic equation

$$\nu^2\delta + 2\nu - \delta = 0$$

which has the solutions

$$\nu^* = \frac{-1 \pm \sqrt{1+\delta^2}}{\delta} \tag{2.4–15}$$

Since ν^* must be in the $(-1, +1)$ interval for $\mathrm{E}\left[Xe^{\nu X}\right]$ and $\mathrm{E}\left[e^{\nu X}\right]$ to converge, the only acceptable solution is

$$\nu^* = \frac{-1 + \sqrt{1+\delta^2}}{\delta} \tag{2.4–16}$$

Finally, we evaluate the upper bound in Equation 2.4–8 by substituting for ν^* from Equation 2.4–16. The result is

$$\mathrm{P}\left[X \geq \delta\right] \leq \frac{\delta^2}{2(-1+\sqrt{1+\delta^2})} e^{1-\sqrt{1+\delta^2}} \tag{2.4–17}$$

For $\delta \gg 1$, Equation 2.4–17 reduces to

$$P(X \geq \delta) \leq \frac{\delta}{2} e^{-\delta} \tag{2.4–18}$$

We note that the Chernov bound decreases exponentially as δ increases. Consequently, it approximates closely the exact tail probability given by Equation 2.4–13.

EXAMPLE 2.4–2. In performance analysis of communication systems over fading channels, we encounter random variables of the form

$$X = d^2R^2 + 2RdN \tag{2.4–19}$$

where d is a constant, R is a Ricean random variable with parameters s and σ representing channel attenuation due to fading, and N is a zero-mean Gaussian random variable with variance $\frac{N_0}{2}$ representing channel noise. It is assumed that R and N are independent random variables. We are interested to apply the Chernov bounding technique to find an upper bound on $\mathrm{P}\left[X < 0\right]$. From the Chernov bound given in Equation 2.4–5, we have

$$\mathrm{P}\left[X \leq 0\right] \leq \mathrm{E}\left[e^{\nu X}\right], \quad \text{for all } \nu < 0 \tag{2.4–20}$$

To determine $\mathrm{E}\left[e^{\nu X}\right]$, we use the well-known relation

$$\mathrm{E}\left[Y\right] = \mathrm{E}\left[\mathrm{E}\left[Y|X\right]\right] \tag{2.4–21}$$

from elementary probability. We note that conditioned on R, X is a Gaussian random variable with mean d^2R^2 and variance $2R^2d^2N_0$. Using the relation for the moment generating function of a Gaussian random variable from Table 2.3–3, we have

$$\mathrm{E}\left[e^{\nu X}|R\right] = e^{\nu d^2R^2+\nu^2d^2N_0R^2} = e^{\nu d^2(1+N_0\nu)R^2} \tag{2.4–22}$$

Now noting that R^2 is a noncentral χ^2 random variable with two degrees of freedom, and using the characteristic function for this random variable from Table 2.3–3, we obtain

$$\begin{aligned} \mathrm{E}\left[e^{\nu X}\right] &= \mathrm{E}\left[\mathrm{E}\left[e^{\nu X}|R\right]\right] \\ &= \mathrm{E}\left[e^{\nu d^2(1+N_0\nu)R^2}\right] \\ &= \frac{1}{1-2\nu d^2(1+N_0\nu)\sigma^2}\, e^{\frac{\nu d^2(1+N_0\nu)s^2}{1-2\nu d^2(1+N_0\nu)\sigma^2}} \end{aligned} \tag{2.4–23}$$

where we have used Equation 2.4–21. From Equations 2.4–20 and 2.4–23 we conclude that

$$\mathrm{P}[X \le 0] \le \min_{\nu<0} \frac{1}{1-2\nu d^2(1+N_0\nu)\sigma^2}\, e^{\frac{\nu d^2(1+N_0\nu)s^2}{1-2\nu d^2(1+N_0\nu)\sigma^2}} \tag{2.4–24}$$

It can be easily verified by differentiation that in the range of interest ($\nu < 0$), the right-hand side is an increasing function of $\lambda = \nu d^2(1+N_0\nu)$, and therefore the minimum is achieved when λ is minimized. By simple differentiation we can verify that λ is minimized for $\nu = -\frac{1}{2N_0}$, resulting in

$$\mathrm{P}[X \le 0] \le \frac{1}{1+\frac{d^2}{2N_0}\sigma^2}\, e^{-\frac{\frac{d^2}{4N_0}s^2}{1+\frac{d^2}{2N_0}\sigma^2}} \tag{2.4–25}$$

If we use Equation 2.3–61 or 2.3–62 for the Ricean random variable, we obtain the following bounds:

$$\mathrm{P}[X \le 0] \le \frac{K+1}{K+1+\frac{A^2d^2}{4N_0}}\, e^{-\frac{\frac{A^2Kd^2}{4N_0}}{K+1+\frac{d^2A^2}{4N_0}}} \tag{2.4–26}$$

and

$$\mathrm{P}[X \le 0] \le \frac{K+1}{K+1+\frac{d^2}{4N_0}}\, e^{-\frac{\frac{Kd^2}{4N_0}}{K+1+\frac{d^2}{4N_0}}} \tag{2.4–27}$$

For the case of Rayleigh fading channels, in which $s = 0$, these relations reduce to

$$\mathrm{P}[X \le 0] \le \frac{1}{1+\frac{d^2}{2N_0}\sigma^2} \tag{2.4–28}$$

Chernov Bound for Sums of Random Variables

Let $\{X_i\}$, $1 \le i \le n$, denote a sequence of iid random variables and define

$$Y = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.4–29}$$

We are interested to find a bound on $\mathrm{P}[Y > \delta]$, where $\delta > \mathrm{E}[X]$. Applying the Chernov bound, we have

$$\begin{aligned} \mathrm{P}[Y > \delta] &= \mathrm{P}\left[\sum_{i=1}^{n} X_i > n\delta\right] \\ &\leq \mathrm{E}\left[e^{\nu\left(\sum_{i=1}^{n} X_i - n\delta\right)}\right] \\ &= \left[\mathrm{E}\left[e^{\nu(X-\delta)}\right]\right]^n, \quad \nu > 0 \end{aligned} \tag{2.4–30}$$

To find the optimal choice of ν we equate the derivative of the right-hand side to zero

$$\frac{d}{d\nu}\left[\mathrm{E}\left[e^{\nu(X-\delta)}\right]\right]^n = n\left[\mathrm{E}\left[e^{\nu(X-\delta)}\right]\right]^{n-1}\mathrm{E}\left[(X-\delta)e^{\nu(X-\delta)}\right] = 0 \tag{2.4–31}$$

The single root of this equation is obtained by solving

$$\mathrm{E}\left[Xe^{\nu X}\right] = \delta\,\mathrm{E}\left[e^{\nu X}\right] \tag{2.4–32}$$

which is exactly Equation 2.4–7. Therefore, for the sum of iid random variables we find the ν^* solution of Equation 2.4–7, and then we use

$$\mathrm{P}[Y > \delta] \leq \left[\mathrm{E}\left[e^{\nu^*(X-\delta)}\right]\right]^n = e^{-n\nu^*\delta}\left[\mathrm{E}\left[e^{\nu^* X}\right]\right]^n \tag{2.4–33}$$

EXAMPLE 2.4–3. The X_i's are binary iid random variables with $\mathrm{P}[X=1] = 1 - \mathrm{P}[X=-1] = p$, where $p < \frac{1}{2}$. We are interested to find a bound on

$$\mathrm{P}\left[\sum_{i=1}^{n} X_i > 0\right]$$

We have $\mathrm{E}[X] = p - (1-p) = 2p - 1 < 0$. Assuming $\delta = 0$, the condition $\delta > \mathrm{E}[X]$ is satisfied, and the preceding development can be applied to this case. We have

$$\mathrm{E}\left[Xe^{\nu X}\right] = pe^{\nu} - (1-p)e^{-\nu} \tag{2.4–34}$$

and Equation 2.4–7 becomes

$$pe^{\nu} - (1-p)e^{-\nu} = 0 \tag{2.4–35}$$

which has the unique solution

$$\nu^* = \frac{1}{2}\ln\frac{1-p}{p} \tag{2.4–36}$$

Using this value, we have

$$\mathrm{E}\left[e^{\nu^* X}\right] = p\sqrt{\frac{1-p}{p}} + (1-p)\sqrt{\frac{p}{1-p}} = 2\sqrt{p(1-p)} \tag{2.4–37}$$

Substituting this result into Equation 2.4–33 results in

$$\mathrm{P}\left[\sum_{i=1}^{n} X_i > 0\right] \leq [4p(1-p)]^{\frac{n}{2}} \tag{2.4–38}$$

Since for $p < \frac{1}{2}$ we have $4p(1-p) < 1$, the bound given in Equation 2.4–38 tends to zero exponentially.

2.5
LIMIT THEOREMS FOR SUMS OF RANDOM VARIABLES

If $\{X_i, i = 1, 2, 3, \ldots\}$ represents a sequence of iid random variables, then it is intuitively clear that the running average of this sequence, i.e.,

$$Y_n = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.5–1}$$

should in some sense converge to the average of the random variables. Two limit theorems, i.e., the *law of large numbers* (LLN) and the *central limit theorem* (CLT), rigorously state how the running average of the random variable behaves as n becomes large.

The (strong) law of large numbers states that if $\{X_i, i = 1, 2, \ldots\}$ is a sequence of iid random variables with $\mathrm{E}[X_1] < \infty$, then

$$\frac{1}{n}\sum_{i=1}^{n} X_i \longrightarrow \mathrm{E}[X_1] \tag{2.5–2}$$

where the type of convergence is *convergence almost everywhere* (a.e.) or *convergence almost surely* (a.s.), meaning the set of points in the probability space for which the left-hand side does not converge to the right-hand side has zero probability.

The central limit theorem states that if $\{X_i, i = 1, 2, \ldots\}$ is a sequence of iid random variables with $m = \mathrm{E}[X_1] < \infty$ and $\sigma^2 = \mathrm{VAR}[X_1] < \infty$, then we have

$$\frac{\frac{1}{n}\sum_{i=1}^{n} X_i - m}{\frac{\sigma}{\sqrt{n}}} \longrightarrow \mathcal{N}(0, 1) \tag{2.5–3}$$

The type of convergence in the CLT is *convergence in distribution*, meaning the CDF of the left-hand side converges to the CDF of $\mathcal{N}(0, 1)$ as n increases.

2.6
COMPLEX RANDOM VARIABLES

A complex random variable $Z = X + jY$ can be considered as a pair of real random variables X and Y. Therefore, we treat a complex random variable as a two-dimensional random vector with components X and Y. The PDF of a complex random variable is defined to be the joint PDF of its real and complex parts. If X and Y are jointly Gaussian random variables, then Z is a complex Gaussian random variable. The PDF of a zero-mean complex Gaussian random variable Z with iid real and imaginary parts is given by

$$p(z) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.6–1}$$

$$= \frac{1}{2\pi\sigma^2} e^{-\frac{|z|^2}{2\sigma^2}} \tag{2.6–2}$$

For a complex random variable Z, the mean and variance are defined by

$$\mathrm{E}[Z] = \mathrm{E}[X] + j\,\mathrm{E}[Y] \tag{2.6–3}$$

$$\mathrm{VAR}[Z] = \mathrm{E}\left[|Z|^2\right] - |\mathrm{E}[Z]|^2 = \mathrm{VAR}[X] + \mathrm{VAR}[Y] \tag{2.6–4}$$

2.6–1 Complex Random Vectors

A complex random vector is defined as $\boldsymbol{Z} = \boldsymbol{X} + j\boldsymbol{Y}$, where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are real-valued random vectors of size n. We define the following real-valued matrices for a complex random vector $\boldsymbol{Z}$.

$$\boldsymbol{C}_X = E\left[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])^t\right] \tag{2.6–5}$$

$$\boldsymbol{C}_Y = E\left[(\boldsymbol{Y} - E[\boldsymbol{Y}])(\boldsymbol{Y} - E[\boldsymbol{Y}])^t\right] \tag{2.6–6}$$

$$\boldsymbol{C}_{XY} = E\left[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{Y} - E[\boldsymbol{Y}])^t\right] \tag{2.6–7}$$

$$\boldsymbol{C}_{YX} = E\left[(\boldsymbol{Y} - E[\boldsymbol{Y}])(\boldsymbol{X} - E[\boldsymbol{X}])^t\right] \tag{2.6–8}$$

Matrices $\boldsymbol{C}_X$ and $\boldsymbol{C}_Y$ are the *covariance matrices* of real random vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively, and hence they are symmetric and nonnegative definite. It is clear from above that $\boldsymbol{C}_{YX} = \boldsymbol{C}_{XY}^t$.

The PDF of $\boldsymbol{Z}$ is the joint PDF of its real and imaginary parts. If we define the $2n$-dimensional real vector

$$\tilde{\boldsymbol{Z}} = \begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \end{bmatrix} \tag{2.6–9}$$

then the PDF of the complex vector $\boldsymbol{Z}$ is the PDF of the real vector $\tilde{\boldsymbol{Z}}$. It is clear that $\boldsymbol{C}_{\tilde{Z}}$, the covariance matrix of $\tilde{\boldsymbol{Z}}$, can be written as

$$\boldsymbol{C}_{\tilde{Z}} = \begin{bmatrix} \boldsymbol{C}_X & \boldsymbol{C}_{XY} \\ \boldsymbol{C}_{YX} & \boldsymbol{C}_Y \end{bmatrix} \tag{2.6–10}$$

We also define the following two, in general complex-valued, matrices

$$\boldsymbol{C}_Z = E\left[(\boldsymbol{Z} - E[\boldsymbol{Z}])(\boldsymbol{Z} - E[\boldsymbol{Z}])^H\right] \tag{2.6–11}$$

$$\widetilde{\boldsymbol{C}}_Z = E\left[(\boldsymbol{Z} - E[\boldsymbol{Z}])(\boldsymbol{Z} - E[\boldsymbol{Z}])^t\right] \tag{2.6–12}$$

where $\boldsymbol{A}^t$ denotes the transpose and $\boldsymbol{A}^H$ denotes the Hermitian transpose of $\boldsymbol{A}$ ($\boldsymbol{A}$ is transposed and each element of it is conjugated). $\boldsymbol{C}_Z$ and $\widetilde{\boldsymbol{C}}_Z$ are called the *covariance* and the *pseudocovariance* of the complex random vector $\boldsymbol{Z}$, respectively. It is easy to verify that for any $\boldsymbol{Z}$, the covariance matrix is Hermitian[†] and nonnegative definite. The pseudocovariance is *skew-Hermitian.*

From these definitions it is easy to verify the following relations.

$$\boldsymbol{C}_Z = \boldsymbol{C}_X + \boldsymbol{C}_Y + j\,(\boldsymbol{C}_{YX} - \boldsymbol{C}_{XY}) \tag{2.6–13}$$

$$\tilde{\boldsymbol{C}}_Z = \boldsymbol{C}_X - \boldsymbol{C}_Y + j\,(\boldsymbol{C}_{XY} + \boldsymbol{C}_{YX}) \tag{2.6–14}$$

$$\boldsymbol{C}_X = \frac{1}{2}\,\mathrm{Re}\,[\boldsymbol{C}_Z + \tilde{\boldsymbol{C}}_Z] \tag{2.6–15}$$

$$\boldsymbol{C}_Y = \frac{1}{2}\,\mathrm{Re}\,[\boldsymbol{C}_Z - \tilde{\boldsymbol{C}}_Z] \tag{2.6–16}$$

$$\boldsymbol{C}_{YX} = \frac{1}{2}\,\mathrm{Im}\,[\boldsymbol{C}_Z + \tilde{\boldsymbol{C}}_Z] \tag{2.6–17}$$

$$\boldsymbol{C}_{XY} = \frac{1}{2}\,\mathrm{Im}\,[\tilde{\boldsymbol{C}}_Z - \boldsymbol{C}_Z] \tag{2.6–18}$$

Proper and Circularly Symmetric Random Vectors

A complex random vector $\boldsymbol{Z}$ is called *proper* if its pseudocovariance is zero, i.e., if $\tilde{\boldsymbol{C}}_Z = \boldsymbol{0}$. From Equation 2.6–14 it is clear that for a proper random vector we have

$$\boldsymbol{C}_X = \boldsymbol{C}_Y \tag{2.6–19}$$

$$\boldsymbol{C}_{XY} = -\boldsymbol{C}_{YX} \tag{2.6–20}$$

Substituting these results into Equations 2.6–13 to 2.6–18 and 2.6–10, we conclude that for proper random vectors

$$\boldsymbol{C}_Z = 2\boldsymbol{C}_X + 2j\boldsymbol{C}_{YX} \tag{2.6–21}$$

$$\boldsymbol{C}_X = \boldsymbol{C}_Y = \frac{1}{2}\,\mathrm{Re}\,[\boldsymbol{C}_Z] \tag{2.6–22}$$

$$\boldsymbol{C}_{YX} = -\boldsymbol{C}_{XY} = \frac{1}{2}\,\mathrm{Im}\,[\boldsymbol{C}_Z] \tag{2.6–23}$$

$$\boldsymbol{C}_{\tilde{Z}} = \begin{bmatrix} \boldsymbol{C}_X & \boldsymbol{C}_{XY} \\ -\boldsymbol{C}_{XY} & \boldsymbol{C}_X \end{bmatrix} \tag{2.6–24}$$

For the special case of $n = 1$, i.e., when we are dealing with a single complex random variable $Z = X + jY$, the conditions for being proper become

$$\mathrm{VAR}\,[X] = \mathrm{VAR}\,[Y] \tag{2.6–25}$$

$$\mathrm{COV}\,[X, Y] = -\mathrm{COV}\,[Y, X] \tag{2.6–26}$$

which means that Z is proper if X and Y have equal variances and are uncorrelated. In this case $\mathrm{VAR}\,[Z] = 2\,\mathrm{VAR}\,[X]$. Since in the case of jointly Gaussian random variables uncorrelated is equivalent to independent, we conclude that a complex Gaussian random variable Z is proper if and only if its real and complex parts are independent with equal variance. For a zero-mean proper complex Gaussian random variable, the PDF is given by Equation 2.6–2.

[†] Matrix $\boldsymbol{A}$ is Hermitian if $\boldsymbol{A} = \boldsymbol{A}^{\mathrm{H}}$. It is skew-Hermitian if $\boldsymbol{A}^{\mathrm{H}} = -\boldsymbol{A}$.

If the complex random vector $\boldsymbol{Z} = \boldsymbol{X} + j\boldsymbol{Y}$ is Gaussian, meaning that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are jointly Gaussian, then we have

$$p(z) = p(\tilde{z}) = \frac{1}{(2\pi)^n (\det \boldsymbol{C}_{\tilde{Z}})^{\frac{1}{2}}} e^{-\frac{1}{2}(\tilde{z}-\tilde{\boldsymbol{m}})^t \boldsymbol{C}_{\tilde{Z}}{}^{-1}(\tilde{z}-\tilde{\boldsymbol{m}})} \tag{2.6–27}$$

where

$$\tilde{\boldsymbol{m}} = \mathrm{E}\left[\tilde{\boldsymbol{Z}}\right] \tag{2.6–28}$$

It can be shown that in the special case where $\boldsymbol{Z}$ is a proper n-dimensional complex Gaussian random vector, with mean $\boldsymbol{m} = \mathrm{E}[\boldsymbol{Z}]$ and nonsingular covariance matrix $\boldsymbol{C}_Z$, its PDF can be written as

$$p(z) = \frac{1}{\pi^n \det \boldsymbol{C}_Z} e^{-\frac{1}{2}(z-\boldsymbol{m})^\dagger \boldsymbol{C}_Z{}^{-1}(z-\boldsymbol{m})} \tag{2.6–29}$$

A complex random vector $\boldsymbol{Z}$ is called *circularly symmetric* or *circular* if rotating the vector by any angle does not change its PDF. In other words, a complex random vector $\boldsymbol{Z}$ is circularly symmetric if $\boldsymbol{Z}$ and $e^{j\theta}\boldsymbol{Z}$ have the same PDF for all θ. In Problem 2.34 we will see that if $\boldsymbol{Z}$ is circular, then it is zero-mean and proper, i.e., $\mathrm{E}[\boldsymbol{Z}] = \boldsymbol{0}$ and $\mathrm{E}\left[\boldsymbol{Z}\boldsymbol{Z}^t\right] = \boldsymbol{0}$. In Problem 2.35 we show that if $\boldsymbol{Z}$ is a zero-mean proper Gaussian complex vector, then $\boldsymbol{Z}$ is circular. In other words, *for complex Gaussian random vectors being zero-mean and proper is equivalent to being circular*.

In Problem 2.36 we show that if $\boldsymbol{Z}$ is a proper complex vector, then any *affine transformation* of it, i.e., any transform of the form $\boldsymbol{W} = \boldsymbol{A}\boldsymbol{Z} + \boldsymbol{b}$, is also a proper complex vector. Since we know that if $\boldsymbol{Z}$ is Gaussian, so is $\boldsymbol{W}$, we conclude that if $\boldsymbol{Z}$ is a proper Gaussian vector, so is $\boldsymbol{W}$. For more details on properties of proper and circular random variables and random vectors, the reader is referred to Neeser and Massey (1993) and Eriksson and Koivunen (2006).

2.7

RANDOM PROCESSES

Random processes, stochastic processes, or random signals are fundamental in the study of communication systems. Modeling information sources and communication channels requires a good understanding of random processes and techniques for analyzing them. We assume that the reader has a knowledge of the basic concepts of random processes including definitions of mean, autocorrelation, cross-correlation, stationarity, and ergodicity as given in standard texts such as Leon-Garcia (1994), Papoulis and Pillai (2002), Stark and Woods (2002). In the following paragraphs we present a brief review of the most important properties of random processes.

The mean $m_X(t)$ and the *autocorrelation function* of a random process $X(t)$ are defined as

$$m_X(t) = \mathrm{E}[X(t)] \tag{2.7–1}$$

$$R_X(t_1, t_2) = \mathrm{E}\left[X(t_1)X^*(t_2)\right] \tag{2.7–2}$$

The *cross-correlation function* of two random processes $X(t)$ and $Y(t)$ is defined by

$$R_{XY}(t_1, t_2) = \mathrm{E}\left[X(t_1)Y^*(t_2)\right] \tag{2.7–3}$$

Note that $R_X(t_2, t_1) = R_X^*(t_1, t_2)$, i.e., $R_X(t_1, t_2)$ is Hermitian. For the cross-correlation we have $R_{YX}(t_2, t_1) = R_{XY}^*(t_1, t_2)$.

2.7–1 Wide-Sense Stationary Random Processes

Random process $X(t)$ is *wide-sense stationary* (WSS) if its mean is constant and $R_X(t_1, t_2) = R_X(\tau)$ where $\tau = t_1 - t_2$. For WSS processes $R_X(-\tau) = R_X^*(\tau)$. Two processes $X(t)$ and $Y(t)$ are *jointly wide-sense stationary* if both $X(t)$ and $Y(t)$ are WSS and $R_{XY}(t_1, t_2) = R_{XY}(\tau)$. For jointly WSS processes $R_{YX}(-\tau) = R_{XY}^*(\tau)$. A complex process is WSS if its real and imaginary parts are jointly WSS.

The *power spectral density* (PSD) or *power spectrum* of a WSS random process $X(t)$ is a function $\mathcal{S}_X(f)$ describing the distribution of power as a function of frequency. The unit for power spectral density is watts per hertz. The *Wiener-Khinchin theorem* states that for a WSS process, the power spectrum is the Fourier transform of the autocorrelation function $R_X(\tau)$, i.e.,

$$\mathcal{S}_X(f) = \mathscr{F}[R_X(\tau)] \tag{2.7–4}$$

Similarly, the *cross spectral density* (CSD) of two jointly WSS processes is defined as the Fourier transform of their cross-correlation function.

$$\mathcal{S}_{XY}(f) = \mathscr{F}[R_{XY}(\tau)] \tag{2.7–5}$$

The cross spectral density satisfies the following symmetry property:

$$\mathcal{S}_{XY}(f) = \mathcal{S}_{YX}^*(f) \tag{2.7–6}$$

From properties of the autocorrelation function it is easy to verify that the power spectral density of any real WSS process $X(t)$ is a real, nonnegative, and even function of f. For complex processes, power spectrum is real and nonnegative, but not necessarily even. The cross spectral density can be a complex function, even when both $X(t)$ and $Y(t)$ are real processes.

If $X(t)$ and $Y(t)$ are jointly WSS random processes, then $Z(t) = aX(t) + bY(t)$ is a WSS random process with autocorrelation and power spectral density given by

$$R_Z(\tau) = |a|^2 R_X(\tau) + |b|^2 R_Y(\tau) + ab^* R_{XY}(\tau) + ba^* R_{YX}(\tau) \tag{2.7–7}$$

$$\mathcal{S}_Z(f) = |a|^2 \mathcal{S}_X(f) + |b|^2 \mathcal{S}_Y(f) + 2\,\mathrm{Re}\left[ab^* \mathcal{S}_{XY}(f)\right] \tag{2.7–8}$$

In the special case where $a = b = 1$, we have $Z(t) = X(t) + Y(t)$, which results in

$$R_Z(\tau) = R_X(\tau) + R_Y(\tau) + R_{XY}(\tau) + R_{YX}(\tau) \tag{2.7–9}$$

$$\mathcal{S}_Z(f) = \mathcal{S}_X(f) + \mathcal{S}_Y(f) + 2\,\mathrm{Re}\,[\mathcal{S}_{XY}(f)] \tag{2.7–10}$$

and when $a = 1$ and $b = j$, we have $Z(t) = X(t) + jY(t)$ and

$$R_Z(\tau) = R_X(\tau) + R_Y(\tau) + j\,(R_{YX}(\tau) + R_{XY}(\tau)) \tag{2.7–11}$$

$$\mathcal{S}_Z(f) = \mathcal{S}_X(f) + \mathcal{S}_Y(f) + 2\,\mathrm{Im}\,[\mathcal{S}_{XY}(f)] \tag{2.7–12}$$

When a WSS process $X(t)$ passes through an LTI system with impulse response $h(t)$ and transfer function $H(f) = \mathscr{F}[h(t)]$, the output process $Y(t)$ and $X(t)$ are jointly WSS and the following relations hold:

$$m_Y = m_X \int_{-\infty}^{\infty} h(t)\,dt \tag{2.7–13}$$

$$R_{XY}(\tau) = R_X(\tau) \star h^*(-\tau) \tag{2.7–14}$$

$$R_Y(\tau) = R_X(\tau) \star h(\tau) \star h^*(-\tau) \tag{2.7–15}$$

$$m_Y = m_X H(0) \tag{2.7–16}$$

$$\mathcal{S}_{XY}(f) = \mathcal{S}_X(f) H^*(f) \tag{2.7–17}$$

$$\mathcal{S}_Y(f) = \mathcal{S}_X(f) |H(f)|^2 \tag{2.7–18}$$

The power in a WSS process $X(t)$ is the sum of the powers at all frequencies, and therefore it is the integral of the power spectrum over all frequencies. We can write

$$P_X = \mathrm{E}\left[|X(t)|^2\right] = R_X(0) = \int_{-\infty}^{\infty} \mathcal{S}_X(f)\,df \tag{2.7–19}$$

Gaussian Random Processes

A real random process $X(t)$ is Gaussian if for all positive integers n and for all $(t_1, t_2, \ldots, t_n)$, the random vector $(X(t_1), X(t_2), \ldots, X(t_n))^t$ is a Gaussian random vector; i.e., random variables $\{X(t_i)\}_{i=1}^n$ are jointly Gaussian random variables. Similar to jointly Gaussian random variables, linear filtering of Gaussian random processes results in a Gaussian random process, even when the filtering is time-varying.

Two real random processes $X(t)$ and $Y(t)$ are jointly Gaussian if for all positive integers n, m and all $(t_1, t_2, \ldots, t_n)$, and $(t'_1, t'_2, \ldots, t'_m)$, the random vector

$$(X(t_1), X(t_2), \ldots, X(t_n), Y(t'_1), Y(t'_2), \ldots, Y(t'_m))^t$$

is a Gaussian vector. For two jointly Gaussian random processes $X(t)$ and $Y(t)$, being uncorrelated, i.e., having

$$R_{XY}(t + \tau, t) = \mathrm{E}\,[X(t + \tau)]\,\mathrm{E}\,[Y(t)] \quad \text{for all } t \text{ and } \tau \tag{2.7–20}$$

is equivalent to being independent.

A complex process $Z(t) = X(t) + jY(t)$ is Gaussian if $X(t)$ and $Y(t)$ are jointly Gaussian processes.

White Processes

A process is called a *white process* if its power spectral density is constant for all frequencies; this constant value is usually denoted by $\frac{N_0}{2}$.

$$\mathcal{S}_X(f) = \frac{N_0}{2} \tag{2.7–21}$$

Using Equation 2.7–19, we see that the power in a white process is infinite, indicating that white processes cannot exist as a physical process. Although white processes are not physically realizable processes, they are very useful, closely modeling some important physical phenomenon including the *thermal noise*.

Thermal noise is the noise generated in electric devices by thermal agitation of electrons. Thermal noise can be closely modeled by a random process $N(t)$ having the following properties:

1. $N(t)$ is a stationary process.
2. $N(t)$ is a zero-mean process.
3. $N(t)$ is a Gaussian process.
4. $N(t)$ is a white process whose power spectral density is given by

$$\mathcal{S}_N(f) = \frac{N_0}{2} = \frac{kT}{2} \tag{2.7–22}$$

where T is the ambient temperature in kelvins and k is *Boltzmann's constant*, equal to 38×10^{-23} J/K.

Discrete-Time Random Processes

Discrete-time random processes have similar properties to continuous time processes. In particular the PSD of a WSS discrete-time random process is defined as the discrete-time Fourier transform of its autocorrelation function

$$\mathcal{S}_X(f) = \sum_{m=-\infty}^{\infty} R_X(m) e^{-j2\pi f m} \tag{2.7–23}$$

and the autocorrelation function can be obtained as the inverse Fourier transform of the power spectral density as

$$R_X(m) = \int_{-1/2}^{1/2} \mathcal{S}_X(f) e^{j2\pi f m}\, df \tag{2.7–24}$$

The power in a discrete-time random process is given by

$$P = \mathrm{E}\left[|X(n)|^2\right] = R_X(0) = \int_{-1/2}^{1/2} \mathcal{S}_X(f)\, df \tag{2.7–25}$$

2.7–2 Cyclostationary Random Processes

A random process $X(t)$ is *cyclostationary* if its mean and autocorrelation function are periodic functions with the same period T_0. For a cyclostationary process we have

$$m_X(t + T_0) = m_X(t) \tag{2.7–26}$$

$$R_X(t_1 + T_0, t_2 + T_0) = R_X(t_1, t_2) \tag{2.7–27}$$

Cyclostationary processes are encountered frequently in the study of communication systems because many modulated processes can be modeled as cyclostationary processes. For a cyclostationary process, the average autocorrelation function is defined as the average of the autocorrelation function over one period

$$\overline{R_X(\tau)} = \frac{1}{T_0}\int_0^{T_0} R_X(t + \tau, t)\,dt \tag{2.7–28}$$

The (average) power spectral density for a cyclostationary process is defined as the Fourier transform of the average autocorrelation function, i.e.,

$$\mathcal{S}_X(f) = \mathscr{F}\left[\overline{R_X(\tau)}\right] \tag{2.7–29}$$

EXAMPLE 2.7–1. Let $\{a_n\}$ denote a discrete-time WSS random process with mean $m_a(n) = \mathrm{E}[a_n] = m_a$ and autocorrelation function $R_a(m) = \mathrm{E}\left[a_{n+m}a_n^*\right]$. Define the random process

$$X(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT) \tag{2.7–30}$$

for an arbitrary deterministic function $g(t)$. We have

$$m_X(t) = \mathrm{E}[X(t)] = m_a \sum_{n=-\infty}^{\infty} g(t - nT) \tag{2.7–31}$$

This function is obviously periodic with period T. For the autocorrelation function we have

$$R_X(t + \tau, t) = \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \mathrm{E}\left[a_n a_m^*\right] g(t + \tau - nT)g^*(t - mT) \tag{2.7–32}$$

$$= \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} R_a(n - m)g(t + \tau - nT)g^*(t - mT) \tag{2.7–33}$$

It can readily be verified that

$$R_X(t + \tau + T, t + T) = R_X(t + \tau, t) \tag{2.7–34}$$

Equations 2.7–31 and 2.7–34 show that $X(t)$ is a cyclostationary process.

2.7–3 Proper and Circular Random Processes

For a complex random process $Z(t) = X(t) + jY(t)$, we define the covariance and the pseudocovariance, similar to the case of complex random vectors, as

$$C_Z(t+\tau, t) = \mathrm{E}\left[Z(t+\tau)Z^*(t)\right] \tag{2.7–35}$$

$$\widetilde{C}_Z(t+\tau, t) = \mathrm{E}\left[Z(t+\tau)Z(t)\right] \tag{2.7–36}$$

It is easy to verify that similar to Equations 2.6–13 and 2.6–14, we have

$$C_Z(t+\tau, t) = C_X(t+\tau, t) + C_Y(t+\tau, t) + j\left(C_{YX}(t+\tau, t) - C_{XY}(t+\tau, t)\right) \tag{2.7–37}$$

$$\widetilde{C}_Z(t+\tau, t) = C_X(t+\tau, t) - C_Y(t+\tau, t) + j\left(C_{YX}(t+\tau, t) + C_{XY}(t+\tau, t)\right) \tag{2.7–38}$$

A complex random process $Z(t)$ is *proper* if its pseudocovariance is zero, i.e., $\widetilde{C}_Z(t+\tau, t) = 0$. For a proper random process we have

$$C_X(t+\tau, t) = C_Y(t+\tau, t) \tag{2.7–39}$$

$$C_{YX}(t+\tau, t) = -C_{XY}(t+\tau, t) \tag{2.7–40}$$

and

$$C_Z(t+\tau, t) = 2C_X(t+\tau, t) + j2C_{YX}(t+\tau, t) \tag{2.7–41}$$

If $Z(t)$ is a zero-mean process, then all covariances in Equations 2.7–35 to 2.7–41 are substituted with auto- or cross-correlations. When $Z(t)$ is WSS, all auto- and cross-correlations are functions of τ only. A proper Gaussian random process is a random process for which, for all n and all $(t_1, t_2, \ldots, t_n)$, the complex random vector $(Z(t_1), Z(t_2), \ldots, Z(t_n))^t$ is a proper Gaussian vector.

A complex random process $Z(t)$ is circular if for all θ, $Z(t)$ and $e^{j\theta}Z(t)$ have the same statistical properties. Similar to the case of complex vectors, it can be shown that if $Z(t)$ is circular, then it is both proper and zero-mean. For the case of Gaussian processes, being proper and zero-mean is equivalent to being circular. Also similar to the case of complex vectors, passing a circular Gaussian process through a linear (not necessarily time-invariant) system results in a circular Gaussian process at the output.

2.7–4 Markov Chains

Markov chains are discrete-time, discrete-valued random processes in which the current value depends on the entire past values only through the most recent values. In a jth-order Markov chain, the current value depends on the past values only through the most recent j values, i.e.,

$$\begin{aligned} &\mathrm{P}\left[X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, \ldots\right] \\ &\quad = \mathrm{P}\left[X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, \ldots, X_{n-j} = x_{n-j}\right] \end{aligned} \tag{2.7–42}$$

It is convenient to consider the set of the most recent j values as the *state* of the Markov chain. With this definition the current state of the Markov chain, i.e., $S_n = (X_n, X_{n-1}, \ldots, X_{n-j+1})$, depends only on the most recent state $S_{n-1} = (X_{n-1}, X_{n-2}, \ldots, X_{n-j})$. That is,

$$\mathrm{P}\left[S_n = s_n \,|S_{n-1} = s_{n-1}, S_{n-2} = s_{n-2}, \ldots\right] = \mathrm{P}\left[S_n = s_n \,|S_{n-1} = s_{n-1}\right] \quad (2.7\text{–}43)$$

which represents a first-order Markov chain in terms of the state variable S_n. Note that with this notation, X_n is a deterministic function of state S_n. We can generalize this notion to the case where the state evolves according to Equation 2.7–43 but the output—or the value of the random process X_n—depends on state S_n through a conditional probability mass function

$$\mathrm{P}\left[X_n = x_n \,|S_n = s_n\right] \quad (2.7\text{–}44)$$

With this background, we define a Markov chain[†] as a finite-state machine with state at time n, denoted by S_n, taking values in the set $\{1, 2, \ldots, S\}$ such that Equation 2.7–43 holds and the value of the random process at time n, denoted by X_n and taking values in a discrete set, depends statistically on the state through the conditional PMF $\mathrm{P}\left[X_n = x_n \,|S_n = s_n\right]$.

The internal development of the process depends on the set of states and the probabilistic law that governs the transitions between the states. If $\mathrm{P}\left[S_n \,|S_{n-1}\right]$ is independent of n (time), the Markov chain is called *homogeneous*. In this case the probability of transition from state i to state j, $1 \le i, j \le S$, is independent of n and is denoted by P_{ij}

$$P_{ij} = \mathrm{P}\left[S_n = j \,|S_{n-1} = i\right] \quad (2.7\text{–}45)$$

In a homogeneous Markov chain, we define the *state transition matrix*, or *one-step transition matrix*, $\boldsymbol{P}$ as a matrix with elements P_{ij}. The element at row i and column j denotes the probability of a direct transition from state i to state j. $\boldsymbol{P}$ is a matrix with nonnegative elements, and the sum of each row of it is equal to 1. The n-step transition matrix gives the probabilities of moving from i to j in n steps. For discrete-time homogeneous Markov chains, the n-step transition matrix is equal to $\boldsymbol{P}^n$. All Markov chains studied here are assumed to be homogeneous.

The row vector $\boldsymbol{p}(n) = [p_1(n) \;\; p_2(n) \cdots, p_S(n)]$, where $p_i(n)$ denotes the probability of being in state i at time n, is the *state probability vector* of the Markov chain at time n. From this definition it is clear that

$$\boldsymbol{p}(n) = \boldsymbol{p}(n-1)\boldsymbol{P} \quad (2.7\text{–}46)$$

and

$$\boldsymbol{p}(n) = \boldsymbol{p}(0)\boldsymbol{P}^n \quad (2.7\text{–}47)$$

[†] Strictly speaking, this is the definition of a finite-state Markov chain (FSMC), which is the only class of Markov chains studied in this book.

If $\lim_{n\to\infty} \boldsymbol{P}^n$ exists and all its rows are equal, we denote each row of the limit by $\boldsymbol{p}$, i.e.,

$$\lim_{n\to\infty} \boldsymbol{P}^n = \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{p} \\ \vdots \\ \boldsymbol{p} \end{bmatrix} \tag{2.7–48}$$

In this case

$$\lim_{n\to\infty} \boldsymbol{p}(n) = \lim_{n\to\infty} \boldsymbol{p}(0)\boldsymbol{P}^n = \boldsymbol{p}(0) \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{p} \\ \vdots \\ \boldsymbol{p} \end{bmatrix} = \boldsymbol{p} \tag{2.7–49}$$

This means that starting from *any* initial probability vector $\boldsymbol{p}(0)$, the Markov chain stabilizes at the state probability vector given by $\boldsymbol{p}$, which is called the *steady-state*, *equilibrium*, or *stationary* state probability distribution of the Markov chain. Since after reaching the steady-state probability distribution these probabilities do not change, $\boldsymbol{p}$ can be obtained as the solution of the equation

$$\boldsymbol{p}\boldsymbol{P} = \boldsymbol{p} \tag{2.7–50}$$

that satisfies the conditions $p_i \geq 0$ and $\sum_i p_i = 1$ (i.e., it is a probability vector). If a Markov chain starts from state $\boldsymbol{p}$, then it will always remain in this state, because $\boldsymbol{p}\boldsymbol{P} = \boldsymbol{p}$. Some basic questions are the following: Does $\boldsymbol{p}\boldsymbol{P} = \boldsymbol{p}$ always have a solution that is a probability vector? If yes, under what conditions is this solution unique? Under what conditions does $\lim_{n\to\infty} \boldsymbol{P}^n$ exist? If the limit exists, does the limit have equal rows?

If it is possible to move from any state of a Markov chain to any other state in a finite number of steps, the Markov chain is called *irreducible*. The *period* of state i of a Markov chain is the greatest common divisor (GCD) of all n such that $P_{ii}(n) > 0$. State i is *aperiodic* if its period is equal to 1. A finite-state Markov chain is called *ergodic* if it is irreducible and all its states are aperiodic.

It can be shown that in an ergodic Markov chain $\lim_{n\to\infty} \boldsymbol{P}^n$ always exists and all rows of the limit are equal, i.e., Equation 2.7–48 holds. In this case a unique stationary (steady-state) state probability distribution exists and starting from any initial state probability vector, the Markov chain ends up in the steady-state state probability vector $\boldsymbol{p}$.

EXAMPLE 2.7–2. A Markov chain with four states is described by the finite-state diagram shown in Figure 2.7–1. For this Markov chain we have

$$\boldsymbol{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{3} & 0 & \frac{1}{6} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & \frac{1}{4} & 0 & \frac{3}{4} \\ \frac{5}{6} & 0 & \frac{1}{6} & 0 \end{bmatrix}. \tag{2.7–51}$$

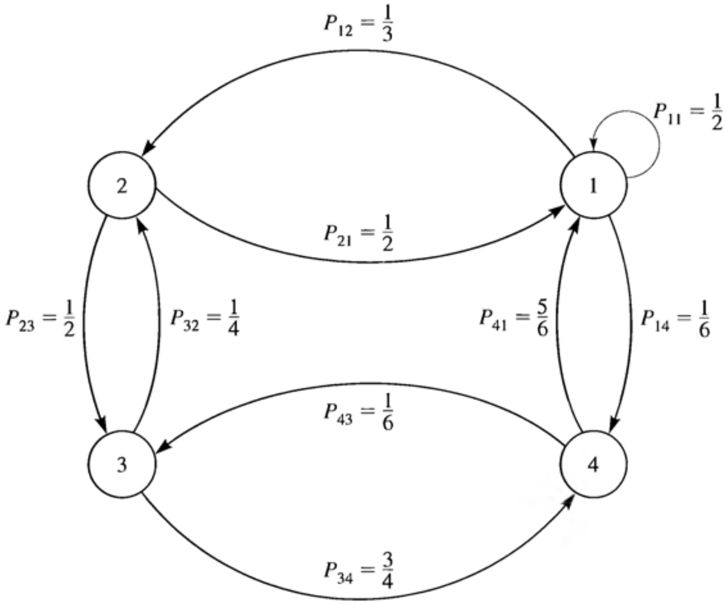


FIGURE 2.7–1
State transition diagram for a FSMC.

It is easily verified that this Markov chain is irreducible and aperiodic, and thus ergodic. To find the steady-state probability distribution, we can either find the limit of $\boldsymbol{P}^n$ as $n \to \infty$ or solve Equation 2.7–50. The result is

$$\boldsymbol{p} \approx [0.49541 \quad 0.19725 \quad 0.12844 \quad 0.17889] \tag{2.7–52}$$

2.8
SERIES EXPANSION OF RANDOM PROCESSES

Series expansion of random processes results in expressing the random processes in terms of a sequence of random variables as coefficients of orthogonal or orthonormal basis functions. This type of expansion reduces working with random processes to working with random variables, which in many cases are easier to handle. In the following we describe two types of series expansions for random processes. First we describe the sampling theorem for band-limited random processes, and then we continue with the Karhunen-Loeve expansion of random processes, which is a more general expansion.

2.8–1 Sampling Theorem for Band-Limited Random Processes

A deterministic real signal $x(t)$ with Fourier transform $X(f)$ is called band-limited if $X(f) = 0$ for $|f| > W$, where W is the highest frequency contained in $x(t)$. Such a signal is uniquely represented by samples of $x(t)$ taken at a rate of $f_s \geq 2W$ samples/s. The minimum rate $f_N = 2W$ samples/s is called the *Nyquist rate*. For complex-valued signals W is one-half of the frequency support of the signal; i.e., if W_1 and W_2

are the lowest and the highest frequency components of the signal, respectively, then $2W = W_2 - W_1$. The signal can be perfectly reconstructed from its sampled values if the sampling rate is at least equal to $2W$. The difference, however, is that the sampled values are complex in this case, and for specifying each sample, two real numbers are required. This means that a real signal can be perfectly described in terms of $2W$ real numbers per second, or it has $2W$ *degrees of freedom* or *real dimensions* per second. For a complex signal the number of degrees of freedom is $4W$ per second, which is equivalent to $2W$ *complex dimensions* or $4W$ real dimensions per second.

Sampling below the Nyquist rate results in frequency aliasing. The band-limited signal sampled at the Nyquist rate can be reconstructed from its samples by use of the interpolation formula

$$x(t) = \sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right) \operatorname{sinc}\left[2W\left(t - \frac{n}{2W}\right)\right] \tag{2.8–1}$$

where $\{x(n/2W)\}$ are the samples of $x(t)$ taken at $t = n/2W$, $n = 0, \pm1, \pm2, \ldots$. Equivalently, $x(t)$ can be reconstructed by passing the sampled signal through an ideal lowpass filter with impulse response $h(t) = \operatorname{sinc}(2Wt)$. Figure 2.8–1 illustrates the signal reconstruction process based on ideal interpolation. Note that the expansion of $x(t)$ as given by Equation 2.8–1 is an orthogonal expansion and not an orthonormal expansion since

$$\int_{-\infty}^{\infty} \operatorname{sinc}\left[2W\left(t - \frac{n}{2W}\right)\right] \operatorname{sinc}\left[2W\left(t - \frac{m}{2W}\right)\right] dt = \begin{cases} \frac{1}{2W} & n = m \\ 0 & n \neq m \end{cases} \tag{2.8–2}$$

A stationary stochastic process $X(t)$ is said to be band-limited if its power spectral density $\mathcal{S}_X(f) = 0$ for $|f| > W$. Since $\mathcal{S}_X(f)$ is the Fourier transform of the autocorrelation function $R_X(\tau)$, it follows that $R_X(\tau)$ can be represented as

$$R_X(\tau) = \sum_{n=-\infty}^{\infty} R_X\left(\frac{n}{2W}\right) \operatorname{sinc}\left[2W\left(\tau - \frac{n}{2W}\right)\right] \tag{2.8–3}$$

where $\{R_X(n/2W)\}$ are samples of $R_X(\tau)$ taken at $\tau = n/2W$, $n = 0, \pm1, \pm2, \ldots$. Now, if $X(t)$ is a band-limited stationary stochastic process, then $X(t)$ can be represented as

$$X(t) = \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right) \operatorname{sinc}\left[2W\left(t - \frac{n}{2W}\right)\right] \tag{2.8–4}$$

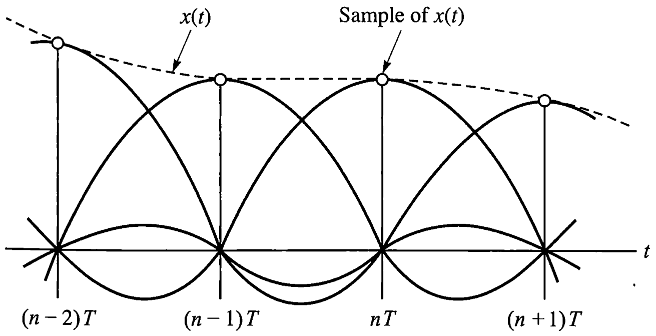


FIGURE 2.8–1
Sampling and reconstruction from samples.

where $\{X(n/2W)\}$ are samples of $X(t)$ taken at $t = n/2W, n = 0, \pm 1, \pm 2, \ldots$. This is the sampling representation for a stationary stochastic process. The samples are random variables that are described statistically by appropriate joint probability density functions. If $X(t)$ is a WSS process, then random variables $\{X(n/2W)\}$ represent a WSS discrete-time random process. The autocorrelation of the sample random variables is given by

$$\begin{aligned} \mathrm{E}\left[X\left(\frac{n}{2W}\right)X^*\left(\frac{m}{2W}\right)\right] &= R_X\left(\frac{n-m}{2W}\right) \\ &= \int_{-W}^{W} \mathcal{S}_X(f)e^{j2\pi f\frac{n-m}{2W}}\,df \end{aligned} \tag{2.8–5}$$

If the process $X(t)$ is filtered white Gaussian noise, then it is zero-mean and its power spectrum is flat in the $[-W, W]$ interval. In this case the samples are uncorrelated, and since they are Gaussian, they are independent as well.

The signal representation in Equation 2.8–4 is easily established by showing that (Problem 2.44)

$$\mathrm{E}\left[\left|X(t) - \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right)\operatorname{sinc}\left[2W\left(t-\frac{n}{2W}\right)\right]\right|^2\right] = 0 \tag{2.8–6}$$

Hence, equality between the sampling representation and the stochastic process $X(t)$ holds in the sense that the mean square error is zero.

2.8–2 The Karhunen-Loève Expansion

The sampling theorem presented above gives a straightforward method for orthogonal expansion of band-limited processes. In this section we present the Karhunen-Loève expansion, an orthonormal expansion that applies to a large class of random processes and results in uncorrelated random variables as expansion coefficients. We present only the results of the Karhunen-Loève expansion. The reader is referred to Van Trees (1968) or Loève (1955) for details.

There are many ways in which a random process can be expanded in terms of a sequence of random variables $\{X_n\}$ and an orthonormal basis $\{\phi_n(t)\}$. However, if we require the additional condition that the random variables X_n be mutually uncorrelated, then the orthonormal bases have to be the solutions of an eigenfunction problem given by an integral equation whose kernel is the autocovariance function of the random process. Solving this integral equation results in the orthonormal basis $\{\phi_n(t)\}$, and projecting the random process on this basis results in the sequence of uncorrelated random variables $\{X_n\}$.

The Karhunen-Loève expansion states that under mild conditions, a random process $X(t)$ with autocovariance function

$$C_X(t_1, t_2) = R_X(t_1, t_2) - m_X(t_1)m_X^*(t_2) \tag{2.8–7}$$

can be expanded over an interval of interest $[a, b]$ in terms of an orthonormal basis $\{\phi_n(t)\}_{n=1}^{\infty}$ such that the coefficients of expansion are uncorrelated. The $\phi_n(t)$'s are solutions (eigenfunctions) of the integral equation

$$\int_a^b C_X(t_1, t_2)\,\phi_n(t_2)\,dt_2 = \lambda_n \phi_n(t_1), \qquad a < t_1 < b \tag{2.8–8}$$

with appropriate normalization such that

$$\int_a^b |\phi_n(t)|^2\,dt = 1 \tag{2.8–9}$$

The Karhunen-Loève expansion is given by

$$\hat{X}(t) = \sum_{n=1}^{\infty} X_n\,\phi_n(t), \qquad a < t < b$$

with the following properties:

1. Random variables X_n denoting the coefficients of the expansion are projections of the random process $X(t)$ on the basis functions, i.e.,

$$X_n = \langle X(t), \phi_n(t)\rangle = \int_a^b X(t)\,\phi_n^*(t)\,dt \tag{2.8–10}$$

2. Random variables X_n are mutually uncorrelated. Moreover, the variance of X_n is λ_n.

$$\mathrm{COV}\,[X_n, X_m] = \begin{cases} \lambda_n & n = m \\ 0 & n \neq m \end{cases} \tag{2.8–11}$$

3. We have

$$E[\hat{X}(t)] = E[X(t)] = m_X(t), \qquad a < t < b \tag{2.8–12}$$

4. $\hat{X}(t)$ is equal to $X(t)$ in the mean square sense

$$\mathrm{E}\,[|X(t) - \hat{X}(t)|^2] = 0, \qquad a < t < b \tag{2.8–13}$$

5. The covariance $C_X(t_1, t_2)$ can be expanded in terms of the bases and the eigenvalues as given in Equation 2.8–14. This is result is known as *Mercer's theorem.*

$$C_X(t_1, t_2) = \sum_{n=1}^{\infty} \lambda_n \phi_n(t_1)\phi_n(t_2), \qquad a < t_1, t_2 < b \tag{2.8–14}$$

6. The eigenfunctions $\{\phi_n(t)\}_{n=1}^{\infty}$ form a complete basis for expansion of all signals $g(t)$ which have finite energy in the interval $[a, b]$. In other words, if $g(t)$ is such that

$$\int_a^b |g(t)|^2\,dt < \infty$$

then we can expand it in terms of $\{\phi_n(t)\}$ as

$$g(t) = \sum_{n=1}^{\infty} g_n \phi_n(t), \qquad a < t < b \tag{2.8–15}$$

where

$$g_n = \langle g(t), \phi_n(t) \rangle = \int_a^b g(t)\phi_n^*(t)\,dt \tag{2.8–16}$$

Equation 2.8–13, which states the Karhunen-Loève expansion, is usually written in the form

$$X(t) = \sum_{n=1}^{\infty} X_n\, \phi_n(t), \qquad a < t < b \tag{2.8–17}$$

where it is understood that the equality is in the mean square sense. The $\{\phi_n(t)\}$ are obtained by solving Equation 2.8–8 and normalizing the solutions, and the coefficients $\{X_n\}$ are obtained by using Equation 2.8–10.

It is worthwhile noting that the Karhunen-Loève expansion applies to both WSS and nonstationary processes. In the special case where the process is zero-mean, the autocovariance function $C_X(t_1, t_2)$ is substituted with the autocorrelation function $R_X(t_1, t_2)$. If the process $X(t)$ is a Gaussian process, $\{X_n\}$ are independent Gaussian random variables.

EXAMPLE 2.8–1. Let $X(t)$ be a zero-mean white process with power spectral density $\frac{N_0}{2}$. To derive the Karhunen-Loève expansion for this process over an arbitrary interval $[a, b]$, we have to solve the integral equation

$$\int_a^b \frac{N_0}{2}\delta(t_1 - t_2)\phi_n(t_2)\,dt_2 = \lambda_n \phi_n(t_1), \qquad a < t_1 < b \tag{2.8–18}$$

where $\frac{N_0}{2}\delta(t_1 - t_2)$ is the autocorrelation function of the white process. Using the sifting property of the impulse function, we have

$$\frac{N_0}{2}\phi_n(t_1) = \lambda_n \phi_n(t_1), \qquad a < t_1 < b \tag{2.8–19}$$

From this equation we see that $\phi_n(t)$ can be any arbitrary function. Therefore, any orthonormal basis can be used for expansion of white processes, and all coefficients of the expansion X_n will have the same variance of $\frac{N_0}{2}$.

2.9
BANDPASS AND LOWPASS RANDOM PROCESSES

In general, bandpass and lowpass random processes can be defined as WSS processes $X(t)$ for which the autocorrelation function $R_X(\tau)$ is either a bandpass or a lowpass signal. Recall that the autocorrelation function is an ordinary deterministic function with a Fourier transform which represents the power spectral density of the random process $X(t)$. Therefore, for a bandpass process the power spectral density is located around frequencies $\pm f_0$, and for lowpass processes the power spectral density is located around zero frequency.

To be more specific, we define a bandpass (or narrowband) process as a real, zero-mean, and WSS random process whose autocorrelation function is a bandpass signal.

Inspired by Equations 2.1–11, we define the *in-phase* and *quadrature* components of a bandpass random process $X(t)$ as

$$\begin{aligned} X_i(t) &= X(t)\cos 2\pi f_0 t + \widehat{X}(t)\sin 2\pi f_0 t \\ X_q(t) &= \widehat{X}(t)\cos 2\pi f_0 t - X(t)\sin 2\pi f_0 t \end{aligned} \tag{2.9–1}$$

We will now show that

1. $X_i(t)$ and $X_q(t)$ are jointly WSS zero-mean random processes.
2. $X_i(t)$ and $X_q(t)$ have the same power spectral density.
3. $X_i(t)$ and $X_q(t)$ are both lowpass processes; i.e., their power spectral density is located around $f = 0$.

We also define the *lowpass equivalent process* $X_l(t)$ as

$$X_l(t) = X_i(t) + jX_q(t) \tag{2.9–2}$$

and we will derive an expression for its autocorrelation function and power spectral density. In addition we will see that $X_l(t)$ is a proper random process.

Since $X(t)$ by assumption is zero-mean, so is $\widehat{X}(t)$, its Hilbert transform. This is obvious since the Hilbert transform is just a filtering operation. From this observation, it is clear that $X_i(t)$ and $X_q(t)$ are both zero-mean processes.

To derive the autocorrelation function of $X_i(t)$, we have

$$\begin{aligned} R_{X_i}(t+\tau, t) &= \mathrm{E}\left[X_i(t+\tau)X_i(t)\right] \\ &= \mathrm{E}\left[(X(t+\tau)\cos 2\pi f_0(t+\tau) + \widehat{X}(t+\tau)\sin 2\pi f_0(t+\tau))\right. \\ &\qquad \left.\times\, (X(t)\cos 2\pi f_0 t + \widehat{X}(t)\sin 2\pi f_0 t)\right] \end{aligned} \tag{2.9–3}$$

Expanding this relation, we have

$$\begin{aligned} R_{X_i}(t+\tau, t) = {} & R_X(\tau)\cos 2\pi f_0(t+\tau)\cos 2\pi f_0 t \\ & + R_{X\widehat{X}}(t+\tau, t)\cos 2\pi f_0(t+\tau)\sin 2\pi f_0 t \\ & + R_{\widehat{X}X}(t+\tau, t)\sin 2\pi f_0(t+\tau)\cos 2\pi f_0 t \\ & + R_{\widehat{X}\widehat{X}}(t+\tau, t)\sin 2\pi f_0(t+\tau)\sin 2\pi f_0 t \end{aligned} \tag{2.9–4}$$

Since the Hilbert transform is the result of passing the process through an LTI system, we conclude that $X(t)$ and $\widehat{X}(t)$ are jointly WSS and therefore all the auto- and cross-correlations in Equation 2.9–4 are functions of τ only. Using Equations 2.7–17 and 2.7–18, we can easily show that (see Problem 2.56)

$$\begin{aligned} R_{X\hat{X}}(\tau) &= -\widehat{R}_X(\tau) \\ R_{\hat{X}X}(\tau) &= \widehat{R}_X(\tau) \\ R_{\hat{X}\hat{X}}(\tau) &= R_X(\tau) \end{aligned} \tag{2.9–5}$$

Substituting these results into Equation 2.9–4 and using standard trigonometric identities yield

$$R_{X_i}(\tau) = R_X(\tau)\cos(2\pi f_0\tau) + \widehat{R}_X(\tau)\sin(2\pi f_0\tau) \tag{2.9–6}$$

Similarly, we can show that

$$R_{X_q}(\tau) = R_{X_i}(\tau) = R_X(\tau)\cos(2\pi f_0\tau) + \widehat{R}_X(\tau)\sin(2\pi f_0\tau) \tag{2.9–7}$$

$$R_{X_iX_q}(\tau) = -R_{X_qX_i}(\tau) = R_X(\tau)\sin(2\pi f_0\tau) - \widehat{R}_X(\tau)\cos(2\pi f_0\tau) \tag{2.9–8}$$

These relations show that $X_i(t)$ and $X_q(t)$ are zero-mean jointly WSS processes with equal autocorrelation functions (and thus equal power spectral densities).

To derive the common power spectral density of $X_i(t)$ and $X_q(t)$ and their cross spectral density, we derive the Fourier transforms of Equations 2.9–7 and 2.9–8. We need to use the modulation property of the Fourier transform and the fact that the Fourier transform of $\widehat{R}_X(\tau)$ is equal to $-j\,\mathrm{sgn}(f)\mathcal{S}_X(f)$. Given these facts, it is straightforward to derive

$$\mathcal{S}_{X_i}(f) = \mathcal{S}_{X_q}(f) = \begin{cases} \mathcal{S}_X(f+f_0) + \mathcal{S}_X(f-f_0) & |f| < f_0 \\ 0 & \text{otherwise} \end{cases} \tag{2.9–9}$$

$$\mathcal{S}_{X_iX_q}(f) = -\mathcal{S}_{X_qX_i}(f) = \begin{cases} j[\mathcal{S}_X(f+f_0) - \mathcal{S}_X(f-f_0)] & |f| < f_0 \\ 0 & \text{otherwise} \end{cases} \tag{2.9–10}$$

Equation 2.9–9 states that the common power spectral density of the in-phase and quadrature components of $X(t)$ is obtained by shifting the power spectral density of $X(t)$ to left and right by f_0 and adding the results and then removing all components outside $[-f_0, f_0]$. This result also shows that both $X_i(t)$ and $X_q(t)$ are lowpass processes. From Equation 2.9–10 we see that if $\mathcal{S}_X(f+f_0) = \mathcal{S}_X(f-f_0)$ for $|f| < f_0$, then $\mathcal{S}_{X_iX_q}(f) = 0$ and consequently, $R_{X_iX_q}(\tau) = 0$. Since $X_i(t)$ and $X_q(t)$ are zero-mean processes, from $R_{X_iX_q}(\tau) = 0$ we conclude that under this condition $X_i(t)$ and $X_q(t)$ are uncorrelated. One of the cases where we have $\mathcal{S}_X(f+f_0) = \mathcal{S}_X(f-f_0)$ for $|f| < 0$ occurs when $\mathcal{S}_X(f)$ is symmetric around f_0, in which case the in-phase and quadrature components will be uncorrelated processes.

We define the complex process $X_l(t) = X_i(t) + jX_q(t)$ as the lowpass equivalent of $X(t)$. Since $X_i(t)$ and $X_q(t)$ are both lowpass processes, we conclude that $X_l(t)$ is also a lowpass process. Comparing Equations 2.9–7 and 2.9–8 with Equations 2.7–39 and 2.7–40, we can conclude that $X_l(t)$ is a proper random process, and therefore, from

Equation 2.7–41, we have

$$R_{X_l}(\tau) = 2R_{X_i}(\tau) + 2jR_{X_qX_i}(\tau) \tag{2.9–11}$$

$$= 2[R_X(\tau) + j\widehat{R}_X(\tau)]e^{-j2\pi f_0 t} \tag{2.9–12}$$

where we have used Equations 2.9–7 and 2.9–8. Comparing Equations 2.9–12 and 2.1–6, we observe that $R_{X_l}(\tau)$ is twice the lowpass equivalent of $R_X(\tau)$. In other words, *the autocorrelation function of the lowpass equivalent process $X_l(t)$ is twice the lowpass equivalent of the autocorrelation function of the bandpass process $X(t)$.*

Taking Fourier transform of both sides of Equation 2.9–12, we obtain

$$\mathcal{S}_{X_l}(f) = \begin{cases} 4\mathcal{S}_X(f + f_0) & |f| < f_0 \\ 0 & \text{otherwise} \end{cases} \tag{2.9–13}$$

and consequently,

$$\mathcal{S}_X(f) = \frac{1}{4}[\mathcal{S}_{X_l}(f - f_0) + \mathcal{S}_{X_l}(-f - f_0)] \tag{2.9–14}$$

We also observe that if $X(t)$ is a Gaussian process, then $X_i(t)$, $X_q(t)$, and $X_l(t)$ will be jointly Gaussian processes; and since $X_l(t)$ is Gaussian, zero-mean, and proper, we conclude that $X_l(t)$ is a circular process as well. In this case if $\mathcal{S}_X(f + f_0) = \mathcal{S}_X(f - f_0)$ for $|f| < f_0$, then $X_i(t)$ and $X_q(t)$ will be independent processes.

EXAMPLE 2.9–1. White Gaussian noise with power spectral density of $\frac{N_0}{2}$ passes through an ideal bandpass filter with transfer function

$$H(f) = \begin{cases} 1 & |f - f_0| < W \\ 0 & \text{otherwise} \end{cases}$$

where $W < f_0$. The output, called *filtered white noise*, is denoted by $X(t)$. This process has a power spectral density of

$$\mathcal{S}_X(f) = \begin{cases} \frac{N_0}{2} & |f - f_0| < W \\ 0 & \text{otherwise} \end{cases}$$

Since $\mathcal{S}_X(f + f_0) = \mathcal{S}_X(f - f_0)$ for $|f| < f_0$, and the process is Gaussian, $X_i(f)$ and $X_q(f)$ are independent lowpass processes. Using Equation 2.9–9, we conclude that

$$\mathcal{S}_{X_i}(f) = \mathcal{S}_{X_q}(f) = \begin{cases} N_0 & |f| < W \\ 0 & \text{otherwise} \end{cases}$$

and from Equation 2.9–13, we conclude that

$$\mathcal{S}_{X_l}(f) = \begin{cases} 2N_0 & |f| < W \\ 0 & \text{otherwise} \end{cases}$$

2.10

BIBLIOGRAPHICAL NOTES AND REFERENCES

In this chapter we have provided a review of basic concepts and definitions in signal analysis, the theory of probability, and stochastic processes. An advanced book on signal analysis that covers most of the material presented here in detail is the book by Franks (1969). The texts by Davenport and Root (1958), Davenport (1970), Papoulis and Pillai (2002), Peebles (1987), Helstrom (1991), Stark and Woods (2002), and Leon-Garcia (1994) provide engineering-oriented treatments of probability and stochastic processes. A more mathematical treatment of probability theory may be found in the text by Loève (1955). Finally, we cite the book by Miller (1964), which treats multidimensional Gaussian distributions.

PROBLEMS

2.1 Prove the following properties of Hilbert transforms:

a. If $x(t) = x(-t)$, then $\hat{x}(t) = -\hat{x}(-t)$.

b. If $x(t) = -x(-t)$, then $\hat{x}(t) = \hat{x}(-t)$.

c. If $x(t) = \cos\omega_0 t$, then $\hat{x}(t) = \sin\omega_0 t$.

d. If $x(t) = \sin\omega_0 t$, then $\hat{x}(t) = -\cos\omega_0 t$.

e. $\hat{\hat{x}}(t) = -x(t)$

f. $\int_{-\infty}^{\infty} x^2(t)\,dt = \int_{-\infty}^{\infty} \hat{x}^2(t)\,dt$

g. $\int_{-\infty}^{\infty} x(t)\hat{x}(t)\,dt = 0$

2.2 Let $x(t)$ and $y(t)$ denote two bandpass signals, and let $x_l(t)$ and $y_l(t)$ denote their lowpass equivalents with respect to some frequency f_0. We know that in general $x_l(t)$ and $y_l(t)$ are complex signals.

1. Show that

$$\int_{-\infty}^{\infty} x(t)y(t)\,dt = \frac{1}{2}\,\mathrm{Re}\left[\int_{-\infty}^{\infty} x_l(t)y_l^*(t)\,dt\right]$$

2. From this conclude that $\mathcal{E}_x = \frac{1}{2}\mathcal{E}_{x_l}$, i.e., the energy in a bandpass signal is one-half the energy in its lowpass equivalent.

2.3 Suppose that $s(t)$ is either a real- or complex-valued signal that is represented as a linear combination of orthonormal functions $\{f_n(t)\}$, i.e.,

$$\hat{s}(t) = \sum_{k=1}^{K} s_k f_k(t)$$

where

$$\int_{-\infty}^{\infty} f_n(t) f_m^*(t)\,dt = \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases}$$

Determine the expressions for the coefficients $\{s_k\}$ in the expansion $\hat{s}_i(t)$ that minimize the energy

$$\mathcal{E}_e = \int_{-\infty}^{\infty} |s(t) - \hat{s}(t)|^2 \, dt$$

and the corresponding residual error $\mathcal{E}_e$.

2.4 Suppose that a set of M signal waveforms $\{s_{lm}(t)\}$ is complex-valued. Derive the equations for the Gram-Schmidt procedure that will result in a set of $N \leq M$ orthonormal signal waveforms.

2.5 Carry out the Gram-Schmidt orthogonalization of the signals in Figure 2.2–1(a) in the order $s_4(t)$, $s_3(t)$, $s_1(t)$, and thus obtain a set of orthonormal functions $\{f_m(t)\}$. Then determine the vector representation of the signals $\{s_n(t)\}$ by using the orthonormal functions $\{f_m(t)\}$. Also determine the signal energies.

2.6 Assuming that the set of signals $\{\phi_{nl}(t), n = 1, \ldots, N\}$ is an orthonormal basis for representation of $\{s_{ml}(t), m = 1, \ldots, M\}$, show that the set of functions given by Equation 2.2–54 constitutes a $2N$ orthonormal basis that is *sufficient* for representation of M bandpass signals given in Equation 2.2–55.

2.7 Show that

$$\tilde{\phi}(t) = -\widehat{\phi}(t)$$

where $\widehat{\phi}(t)$ denotes the Hilbert transform and ϕ and $\tilde{\phi}$ are given by Equation 2.2–54.

2.8 Determine the correlation coefficients ρ_{km} among the four signal waveforms $\{s_i(t)\}$ shown in Figure 2.2–1 and their corresponding Euclidean distances.

2.9 Prove that $s_l(t)$ is generally a complex-valued signal, and give the condition under which it is real. Assume that $s(t)$ is a real-valued bandpass signal.

2.10 Consider the three waveforms $f_n(t)$ shown in Figure P2.10.

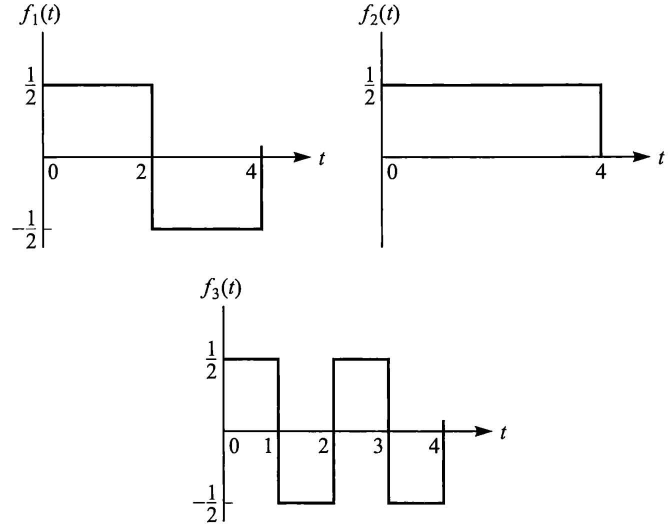


FIGURE P2.10

a. Show that these waveforms are orthonormal.

b. Express the waveform $x(t)$ as a linear combination of $f_n(t),\ n = 1, 2, 3$, if

$$x(t) = \begin{cases} -1 & 0 \le t < 1 \\ 1 & 1 \le t < 3 \\ -1 & 3 \le t < 4 \end{cases}$$

and determine the weighting coefficients.

2.11 Consider the four waveforms shown in Figure P2.11.

a. Determine the dimensionality of the waveforms and a set of basis functions.

b. Use the basis functions to represent the four waveforms by vectors s_1, s_2, s_3, and s_4.

c. Determine the minimum distance between any pair of vectors.

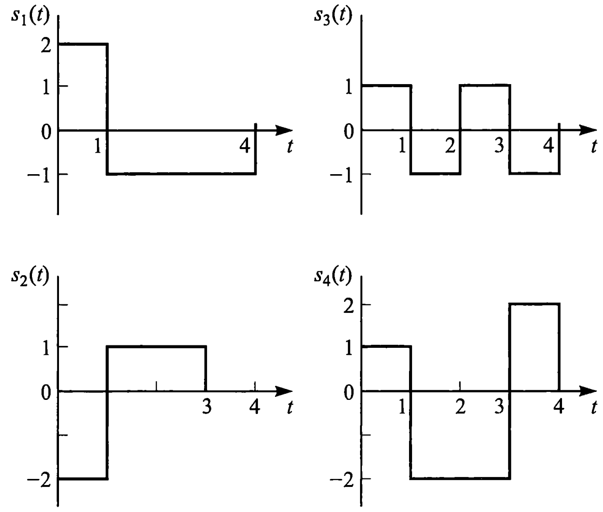


FIGURE P2.11

2.12 Determine a set of orthonormal functions for the four signals shown in Figure P2.12.

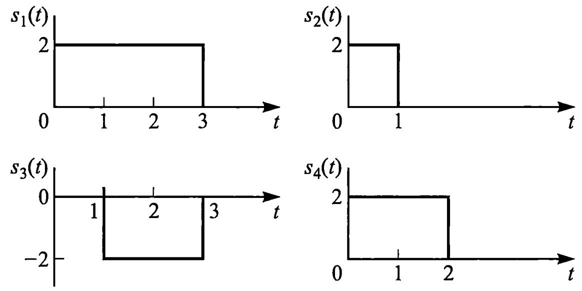


FIGURE P2.12

2.13 A random experiment consists of drawing a ball from an urn that contains 4 red balls numbered 1, 2, 3, 4 and three black balls numbered 1, 2, 3. The following events are defined.

1. E_1 = The number on the ball is even.
2. E_2 = The color of the ball is red, and its number is greater than 1.
3. E_3 = The number on the ball is less than 3.
4. $E_4 = E_1 \cup E_3$
5. $E_5 = E_1 \cup (E_2 \cap E_3)$

Answer the following questions.

1. What is $P(E_2)$?
2. What is $P(E_3|E_2)$?
3. What is $P(E_2|E_4E_3)$?
4. Are E_3 and E_5 independent?

2.14 In a certain city three car brands A, B, C have 20%, 30% and 50% of the market share, respectively. The probability that a car needs major repair during its first year of purchase for the three brands is 5%, 10%, and 15%, respectively.

1. What is the probability that a car in this city needs major repair during its first year of purchase?
2. If a car in this city needs major repair during its first year of purchase, what is the probability that it is made by manufacturer A?

2.15 The random variables X_i, $i = 1, 2, \ldots, n$, have joint PDF $p(x_1, x_2, \ldots, x_n)$. Prove that

$$p(x_1, x_2, x_3, \ldots, x_n) = p(x_n|x_{n-1}, \ldots, x_1)p(x_{n-1}|x_{n-2}, \ldots, x_1) \cdots p(x_3|x_2, x_1)p(x_2|x_1)p(x_1)$$

2.16 A communication channel with binary input and ternary output alphabets is shown in Figure P2.16. The probability of the input being 0 is 0.4. The transition probabilities are shown on the figure.

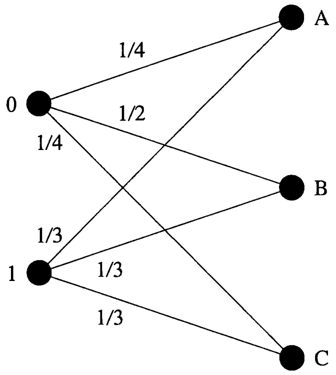


FIGURE P2.16

1. If the channel output is A, what is the best decision on channel input that minimizes the error probability? Repeat for the cases where channel output is B and C.
2. If a 0 is transmitted and an optimal decision scheme (the one derived in part 1) is used at the receiver, what is the probability of error?
3. What is the overall error probability for this channel if the optimal decision scheme is used at the receiver.

2.17 The PDF of a random variable X is $p(x)$. A random variable Y is defined as

$$Y = aX + b$$

where $a < 0$. Determine the PDF of Y in terms of the PDF of X.

2.18 Suppose that X is a Gaussian random variable with zero mean and unit variance. Let

$$Y = aX^3 + b, \qquad a > 0$$

Determine and plot the PDF of Y.

2.19 The noise voltage in an electric circuit can be modeled as a Gaussian random variable with mean equal to zero and variance equal to 10^{-8}.

1. What is the probability that the value of the noise exceeds 10^{-4}? What is the probability that it exceeds 4×10^{-4}? What is the probability that the noise value is between -2×10^{-4} and 10^{-4}?
2. Given that the value of the noise is positive, what is the probability that it exceeds 10^{-4}?

2.20 X is a $\mathcal{N}(0, \sigma^2)$ random variable. This random variable is passed through a system whose input-output relation is given by $y = g(x)$. Find the PDF or the PMF of the output random variable Y in each of the following cases.

1. Square-law device, $g(x) = ax^2$.
2. Limiter,

$$g(x) = \begin{cases} -b & x \leq -b \\ b & x \geq b \\ x & |x| < b \end{cases}$$

3. Hard limiter,

$$g(x) = \begin{cases} a & x > 0 \\ 0 & x = 0 \\ b & x < 0 \end{cases}$$

4. Quantizer, $g(x) = x_n$ for $a_n \leq x < a_{n+1}$, $1 \leq n \leq N$, where x_n lies in the interval $[a_n, a_{n+1}]$ and the sequence $\{a_1, a_2, \ldots, a_{N+1}\}$ satisfies the conditions $a_1 = -\infty$, $a_{N+1} = \infty$ and for $i > j$ we have $a_i > a_j$.

2.21 Shows that for an $\mathcal{N}(m, \sigma^2)$ random variable we have

$$\mathrm{E}\left[(X-m)^n\right] = \begin{cases} 1 \times 3 \times 5 \times \cdots \times (2k-1)\sigma^{2k} = \frac{(2k)!\sigma^{2k}}{2^k k!} & \text{for } n = 2k \\ 0 & \text{for } n = 2k+1 \end{cases}$$

2.22 *a.* Let X_r and X_i be statistically independent zero-mean Gaussian random variables with identical variance. Show that a (rotational) transformation of the form

$$Y_r + jY_i = (X_r + jX_i)e^{j\phi}$$

results in another pair (Y_r, Y_i) of Gaussian random variables that have the same joint PDF as the pair (X_r, X_i).

b. Note that

$$\begin{bmatrix} Y_r \\ Y_i \end{bmatrix} = A \begin{bmatrix} X_r \\ X_i \end{bmatrix}$$

where A is a 2×2 matrix. As a generalization of the two-dimensional transformation of the Gaussian random variables considered in (*a*), what property must the linear transformation A satisfy if the PDFs for X and Y, where $Y = AX$, $X = (X_1 X_2 \cdots X_n)$, and $Y = (Y_1 Y_2 \cdots Y_n)$ are identical?

2.23 Show that if X is a Gaussian vector, the random vector $Y = AX$, where the invertible matrix A represents a linear transformation, is also a Gaussian vector whose mean and covariance matrix are given by

$$\begin{aligned} \boldsymbol{m}_Y &= \boldsymbol{A}\boldsymbol{m}_X \\ \boldsymbol{C}_Y &= \boldsymbol{A}\boldsymbol{C}_X\boldsymbol{A}^t \end{aligned}$$

2.24 The random variable Y is defined as

$$Y = \sum_{i=1}^{n} X_i$$

where the X_i, $i = 1, 2, \ldots, n$, are statistically independent random variables with

$$X_i = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1 - p \end{cases}$$

a. Determine the characteristic function of Y.
b. From the characteristic function, determine the moments $E(Y)$ and $E(Y^2)$.

2.25 This problem provides some useful bounds on $Q(x)$.

1. By integrating $e^{-\frac{u^2+v^2}{2}}$ on the region $u > x$ and $v > x$ in $\mathbb{R}^2$, where $x > 0$, then changing to polar coordinates and upper bounding the integration region by the region $r > \sqrt{2}x$ in the first quadrant, show that $Q(x) \leq \frac{1}{2}e^{-\frac{x^2}{2}}$ for all $x \geq 0$.
2. Apply integration by parts to

$$\int_x^{\infty} e^{-\frac{y^2}{2}} \frac{dy}{y^2}$$

and show that

$$\frac{x}{\sqrt{2\pi}(1+x^2)} e^{-\frac{x^2}{2}} < Q(x) < \frac{1}{\sqrt{2\pi}x} e^{-\frac{x^2}{2}}$$

for all $x > 0$.

3. Based on the result of part 2 show that, for large x,

$$Q(x) \approx \frac{1}{x\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

2.26 Let $X_1, X_2, X_3, \ldots$ denote iid random variables each uniformly distributed on $[0, A]$, where $A > 0$. Let $Y_n = \min\{X_1, X_2, \ldots, X_n\}$.

1. What is the PDF of Y_n?
2. Show that if both A and n go to infinity such that $\frac{n}{A} = \lambda$, where $\lambda > 0$ is a constant, the density function of Y_n tends to an exponential density function. Specify this density function.

2.27 The four random variables X_1, X_2, X_3, X_4 are zero-mean jointly Gaussian random variables with covariance $C_{ij} = E(X_i X_j)$ and characteristic function $\Phi_X(\omega_1, \omega_2, \omega_3, \omega_4)$. Show that

$$E(X_1 X_2 X_3 X_4) = C_{12}C_{34} + C_{13}C_{24} + C_{14}C_{23}$$

2.28 Let

$$\Theta_X(t) = \mathrm{E}\left[e^{tX}\right]$$

denote the moment generating function of random variable X.

1. Using the Chernov bound, show that

$$\ln \mathrm{P}[X \geq \alpha] \leq -\max_{t\geq 0}(\alpha t - \ln \Theta_X(t))$$

2. Define

$$I(\alpha) = \max_{t\geq 0}(\alpha t - \ln \Theta_X(t))$$

as the *large-deviation rate function* of the random variable X, and let $X_1, X_2, \ldots, X_n$ be iid. Define $S_n = (X_1 + X_2 + \cdots + X_n)/n$. Show that for $\alpha \geq \mathrm{E}[X]$

$$\frac{1}{n} \ln \mathrm{P}[S_n \geq \alpha] \leq -I(\alpha)$$

or equivalently

$$\mathrm{P}[S_n \geq \alpha] \leq e^{-nI(\alpha)}$$

Note: It can be shown that for $\alpha \geq \mathrm{E}[X]$, we have $P[S_n \geq \alpha] = e^{-nI(\alpha)+\mathrm{o}(n)}$, where $\mathrm{o}(n) \to 0$ as $n \to \infty$. This result is known as the *large-deviation theorem.*

3. Now assume the X_i's are exponential, i.e.,

$$p_X(x) = \begin{cases} e^{-x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Using the large-deviation result, show that

$$\mathrm{P}[S_n \geq \alpha] = \alpha^n e^{-n(\alpha-1)+\mathrm{o}(n)}$$

for $\alpha \geq 1$.

2.29 From the characteristic functions for the central chi-square and noncentral chi-square random variables given in Table 2.3–3, determine their corresponding first and second moments.

2.30 The PDF of a Cauchy distributed random variable X is

$$p(x) = \frac{a/\pi}{x^2 + a^2}, \qquad -\infty < x < \infty$$

a. Determine the mean and variance of X.
b. Determine the characteristic function of X.

2.31 Let R_0 denote a Rayleigh random variable with PDF

$$f_{R_0}(r_0) = \begin{cases} \frac{r_0}{\sigma^2} e^{-\frac{r_0^2}{2\sigma^2}} & r_0 \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

and R_1 be Ricean with PDF

$$f_{R_1}(r_1) = \begin{cases} \frac{r_1}{\sigma^2} I_0\left(\frac{\mu r_1}{\sigma^2}\right) e^{-\frac{r_1^2+\mu^2}{2\sigma^2}} & r_1 \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Furthermore, assume that R_0 and R_1 are *independent*. Show that

$$P(R_0 > R_1) = \frac{1}{2} e^{-\frac{\mu^2}{4\sigma^2}}$$

2.32 Suppose that we have a complex-valued Gaussian random variable $Z = X + jY$, where (X, Y) are statistically independent variables with zero mean and variance $\mathrm{E}\left[X^2\right] = \mathrm{E}\left[Y^2\right] = \sigma^2$. Let $R = Z + m$, where $m = m_r + jm_i$ and define R as $R = A + jB$. Clearly, $A = X + m_r$ and $B = Y + m_i$. Determine the following probability density functions:

1. $p_{A,B}(a, b)$
2. $p_{U,\Phi}(u, \phi)$, where $U = \sqrt{A^2 + B^2}$ and $\Phi = \tan^{-1} B/A$
3. $p_U(u)$

Note: In part 2 it is convenient to define $\theta = \tan^{-1}(m_i/m_r)$ so that

$$m_r = \sqrt{m_r^2 + m_i^2}\cos\theta, \qquad m_i = \sqrt{m_r^2 + m_i^2}\sin\theta$$

Furthermore, you must use Equation 2.3–34, defining $I_0(\cdot)$ as the modified Bessel function of order zero.

2.33 The random variable Y is defined as

$$Y = \frac{1}{n}\sum_{i=1}^{n} X_i$$

where X_i, $i = 1, 2, \ldots, n$, are statistically independent and identically distributed random variables each of which has the Cauchy PDF given in Problem 2.30.

a. Determine the characteristic function of Y.
b. Determine the PDF of Y.
c. Consider the PDF of Y in the limit as $n \to \infty$. Does the central limit theorem hold? Explain your answer.

2.34 Show that if $\mathbf{Z}$ is circular, then it is zero-mean and proper, i.e., $\mathrm{E}[\mathbf{Z}] = \mathbf{0}$ and $\mathrm{E}\left[\mathbf{Z}\mathbf{Z}^t\right] = \mathbf{0}$.

2.35 Show that if $\mathbf{Z}$ is a zero-mean proper Gaussian complex vector, then $\mathbf{Z}$ is circular.

2.36 Show that if $\mathbf{Z}$ is a proper complex vector, then any transform of the form $\mathbf{W} = \mathbf{A}\mathbf{Z} + \mathbf{b}$ is also a proper complex vector.

2.37 Assume that random processes $X(t)$ and $Y(t)$ are individually and jointly stationary.

a. Determine the autocorrelation function of $Z(t) = X(t) + Y(t)$.
b. Determine the autocorrelation function of $Z(t)$ when $X(t)$ and $Y(t)$ are uncorrelated.
c. Determine the autocorrelation function of $Z(t)$ when $X(t)$ and $Y(t)$ are uncorrelated and have zero means.

2.38 The autocorrelation function of a stochastic process $X(t)$ is

$$R_X(\tau) = \frac{1}{2}N_0\delta(\tau)$$

Such a process is called *white noise*. Suppose $x(t)$ is the input to an ideal bandpass filter having the frequency response characteristic shown in Figure P2.38. Determine the total noise power at the output of the filter.

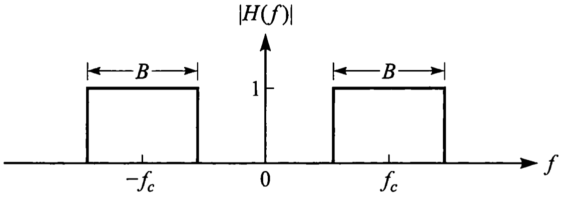


FIGURE P2.38

2.39 A lowpass Gaussian stochastic process $X(t)$ has a power spectral density

$$\mathcal{S}(f) = \begin{cases} N_0 & |f| < B \\ 0 & \text{otherwise} \end{cases}$$

Determine the power spectral density and the autocorrelation function of $Y(t) = X^2(t)$.

2.40 The covariance matrix of three random variables X_1, X_2, and X_3 is

$$\begin{bmatrix} C_{11} & 0 & C_{13} \\ 0 & C_{22} & 0 \\ C_{31} & 0 & C_{33} \end{bmatrix}$$

The linear transformation $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X}$ is made where

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Determine the covariance matrix of $\boldsymbol{Y}$.

2.41 Let $X(t)$ be a stationary real normal process with zero mean. Let a new process $Y(t)$ be defined by

$$Y(t) = X^2(t)$$

Determine the autocorrelation function of $Y(t)$ in terms of the autocorrelation function of $X(t)$. *Hint*: Use the result on Gaussian variables derived in Problem 2.27.

2.42 For the Nakagami PDF, given by Equation 2.3–67, define the normalized random variable $X = R/\sqrt{\Omega}$. Determine the PDF of X.

2.43 The input $X(t)$ in the circuit shown in Figure P2.43 is a stochastic process with $E[X(t)] = 0$ and $R_X(\tau) = \sigma^2\delta(\tau)$; i.e., $X(t)$ is a white noise process.
a. Determine the spectral density $\mathcal{S}_Y(f)$.
b. Determine $R_Y(\tau)$ and $E[Y^2(t)]$.

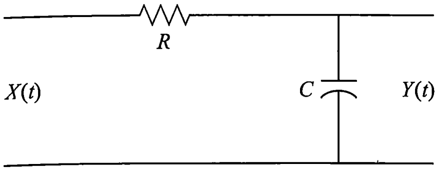


FIGURE P2.43

2.44 Demonstrate the validity of Equation 2.8–6.

2.45 Use the Chernoff bound to show that $Q(x) \le e^{-x^2/2}$.

2.46 Determine the mean, the autocorrelation sequence, and the power density spectrum of the output of a system with unit sample response

$$h(n) = \begin{cases} 1 & n = 0 \\ -2 & n = 1 \\ 1 & n = 2 \\ 0 & \text{otherwise} \end{cases}$$

when the input $x(n)$ is a white noise process with variance σ_x^2.

2.47 The autocorrelation sequence of a discrete-time stochastic process is $R(k) = \left(\frac{1}{2}\right)^{|k|}$. Determine its power density spectrum.

2.48 A discrete-time stochastic process $X(n) \equiv X(nT)$ is obtained by periodic sampling of a continuous-time zero-mean stationary process $X(t)$, where T is the sampling interval; i.e., $f_s = 1/T$ is the sampling rate.

a. Determine the relationship between the autocorrelation function of $X(t)$ and the autocorrelation sequence of $X(n)$.

b. Express the power density spectrum of $X(n)$ in terms of the power density spectrum of the process $X(t)$.

c. Determine the conditions under which the power density spectrum of $X(n)$ is equal to the power density spectrum of $X(t)$.

2.49 The random process $V(t)$ is defined as

$$V(t) = X \cos 2\pi f_c t - Y \sin 2\pi f_c t$$

where X and Y are random variables. Show that $V(t)$ is wide-sense stationary if and only if $E(X) = E(Y) = 0$, $E(X^2) = E(Y^2)$, and $E(XY) = 0$.

2.50 Consider a band-limited zero-mean stationary stochastic process $X(t)$ with power density spectrum

$$\mathcal{S}_X(f) = \begin{cases} 1 & |f| \le W \\ 0 & \text{otherwise} \end{cases}$$

$X(t)$ is sampled at a rate $f_s = 1/T$ to yield a discrete-time process $X(n) \equiv X(nT)$.

a. Determine the expression for the autocorrelation sequence of $X(n)$.

b. Determine the minimum value of T that results in a white (spectrally flat) sequence.

c. Repeat (b) if the power density spectrum of $X(t)$ is

$$\mathcal{S}_X(f) = \begin{cases} 1 - |f|/W & |f| \le W \\ 0 & \text{otherwise} \end{cases}$$

2.51 Show that the functions

$$f_k(t) = \operatorname{sinc}\left[2W\left(t - \frac{k}{2W}\right)\right], \qquad k = 0, \pm 1, \pm 2, \ldots$$

are orthogonal over the real line, i.e.,

$$\int_{-\infty}^{\infty} f_k(t) f_j(t)\,dt = \begin{cases} 1/2W & k = j \\ 0 & \text{otherwise} \end{cases}$$

Therefore, the sampling theorem reconstruction formula may be viewed as a series expansion of the band-limited signal $s(t)$, where the weights are samples of $s(t)$ and the $\{f_k(t)\}$ are the set of orthogonal functions used in the series expansion.

2.52 The noise equivalent bandwidth of a system is defined as

$$B_{\text{eq}} = \frac{1}{G}\int_0^{\infty} |H(f)|^2\,df$$

where $G = \max |H(f)|^2$. Using this definition, determine the noise equivalent bandwidth of the ideal bandpass filter shown in Figure P2.38 and the low-pass system shown in Figure P2.43.

2.53 Suppose that $N(t)$ is a zero-mean stationary narrowband process. The autocorrelation function of the equivalent lowpass process $Z(t) = X(t) + jY(t)$ is defined as

$$R_Z(\tau) = \mathrm{E}\left[Z^*(t)Z(t+\tau)\right]$$

a. Show that

$$\mathrm{E}\left[Z(t)Z(t+\tau)\right] = 0$$

b. Suppose $R_z(\tau) = N_0\delta(\tau)$, and let

$$V = \int_0^T Z(t)\,dt$$

Determine $\mathrm{E}\left[V^2\right]$ and $\mathrm{E}\left[|V|^2\right]$.

2.54 Determine the autocorrelation function of the stochastic process

$$X(t) = A\sin(2\pi f_c t + \Theta)$$

where f_c is a constant and Θ is a uniformly distributed phase, i.e.,

$$p(\theta) = \frac{1}{2\pi}, \qquad 0 \le \theta \le 2\pi$$

2.55 Let $Z(t) = X(t) + jY(t)$ be a complex random process, where $X(t)$ and $Y(t)$ are real-valued, independent, zero-mean, and jointly stationary Gaussian random processes. We assume that $X(t)$ and $Y(t)$ are both band-limited processes with a bandwidth of W and a flat spectral density within their bandwidth, i.e.,

$$\mathcal{S}_X(f) = \mathcal{S}_Y(f) = \begin{cases} N_0 & |f| \le W \\ 0 & \text{otherwise} \end{cases}$$

1. Find $E[Z(t)]$ and $R_Z(t+\tau, t)$, and show that $Z(t)$ is WSS.
2. Find the power spectral density of $Z(t)$.
3. Assume $\phi_1(t), \phi_2(t), \ldots, \phi_n(t)$ are orthonormal, i.e.,

$$\int_{-\infty}^{\infty} \phi_j(t)\phi_k^*(t)\,dt = \begin{cases} 1 & j = k \\ 0 & \text{otherwise} \end{cases}$$

and all $\phi_j(t)$'s are band-limited to $[-W, W]$. Define random variables Z_j as the projections of $Z(t)$ on the $\phi_j(t)$'s, i.e.,

$$Z_j = \int_{-\infty}^{\infty} Z(t)\phi_j^*(t)\,dt, \qquad j = 1, 2, \ldots, n$$

Determine $E[Z_j]$ and $E[Z_j Z_k^*]$ and conclude that the Z_j's are iid zero-mean Gaussian random variables. Find their common variance.

4. Let $Z_j = Z_{jr} + jZ_{ji}$, where Z_{jr} and Z_{ji} denote the real and imaginary parts, respectively, of Z_j. Comment on the joint probability distribution of the $2n$ random variables

$$(Z_{1r}, Z_{1i}, Z_{2r}, Z_{2i}, \ldots, Z_{nr}, Z_{ni})$$

5. Let us define

$$\hat{Z}(t) = Z(t) - \sum_{j=1}^{n} Z_j \phi_j(t)$$

to be the error in expansion of $Z(t)$ as a linear combination of $\phi_j(t)$'s. Show that $E[\hat{Z}(t)Z_k^*] = 0$ for all $k = 1, 2, \ldots, n$. In other words, show that the error $\hat{Z}(t)$ and all the Z_k's are uncorrelated. Can you say $\hat{Z}(t)$ and the Z_k's are independent?

2.56 Let $X(t)$ denote a (real, zero-mean, WSS) bandpass process with autocorrelation function $R_X(\tau)$ and power spectral density $\mathcal{S}_X(f)$, where $\mathcal{S}_X(0) = 0$, and let $\hat{X}(t)$ denote the Hilbert transform of $X(t)$. Then $\hat{X}(t)$ can be viewed as the output of a filter, with impulse response $\frac{1}{\pi t}$ and transfer function $-j\,\mathrm{sgn}(f)$, whose input is $X(t)$. Recall that when $X(t)$ passes through a system with transfer function $H(f)$ and the output is $Y(t)$, we have $\mathcal{S}_Y(f) = \mathcal{S}_X(f)|H(f)|^2$ and $\mathcal{S}_{XY}(f) = \mathcal{S}_X(f)H^*(f)$.

1. Prove that $R_{\hat{X}}(\tau) = R_X(\tau)$.
2. Prove that $R_{X\hat{X}}(\tau) = -\hat{R}_X(\tau)$
3. If $Z(t) = X(t) + j\hat{X}(t)$, determine $\mathcal{S}_Z(f)$.
4. Define $X_l(t) = Z(t)e^{-j2\pi f_0 t}$. Show that $X_l(t)$ is a lowpass WSS random process, and determine $\mathcal{S}_{X_l}(f)$. From the expression for $\mathcal{S}_{X_l}(f)$, derive an expression for $R_{X_l}(\tau)$.

2.57 A noise process has a power spectral density given by

$$\mathcal{S}_n(f) = \begin{cases} 10^{-8}\left(1 - \frac{|f|}{10^8}\right) & |f| < 10^8 \\ 0 & |f| > 10^8 \end{cases}$$

This noise is passed through an ideal bandpass filter with a bandwidth of 2 MHz centered at 50 MHz.

1. Find the power content of the output process.
2. Write the output process in terms of the in-phase and quadrature components, and find the power in each component. Assume $f_0 = 50$ MHz.
3. Find the power spectral density of the in-phase and quadrature components.
4. Now assume that the filter is not an ideal filter and is described by

$$|H(f)|^2 = \begin{cases} \frac{|f|}{10^6} - 49 & 49\text{ MHz} < |f| < 51\text{ MHz} \\ 0 & \text{otherwise} \end{cases}$$

Repeat parts 1, 2, and 3 with this assumption.

3

Digital Modulation Schemes

The digital data are usually in the form of a stream of binary data, i.e., a sequence of 0s and 1s. Regardless of whether these data are inherently digital (for instance, the output of a computer terminal generating ASCII code) or the result of analog-to-digital conversion of an analog source (for instance, digital audio and video), the goal is to reliably transmit these data to the destination by using the given communication channel. Depending on the nature of the communication channel, data can suffer from one or more of certain channel impairments including noise, attenuation, distortion, fading, and interference. To transmit the binary stream over the communication channel, we need to generate a signal that represents the binary data stream and matches the characteristics of the channel. This signal should represent the binary data, meaning that we should be able to retrieve the binary stream from the signal; and it should match the characteristics of the channel, meaning that its bandwidth should match the bandwidth of the channel, and it should be able to resist the impairments caused by the channel. Since different channels cause different types of impairments, signals designed for these channels can be drastically different. The process of mapping a digital sequence to signals for transmission over a communication channel is called *digital modulation* or *digital signaling*. In the process of modulation, usually the transmitted signals are bandpass signals suitable for transmission in the bandwidth provided by the communication channel. In this chapter we study the most commonly used modulation schemes and their properties.

3.1
REPRESENTATION OF DIGITALLY MODULATED SIGNALS

The mapping between the digital sequence (which we assume to be a binary sequence) and the signal sequence to be transmitted over the channel can be either *memoryless* or *with memory*, resulting in memoryless modulation schemes and modulation schemes with memory. In a memoryless modulation scheme, the binary sequence is parsed into subsequences each of length k, and each sequence is mapped into one of the $s_m(t)$,

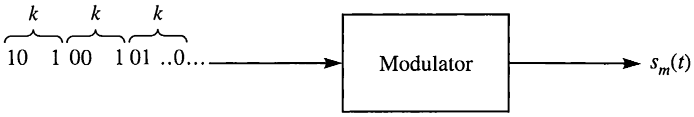


FIGURE 3.1–1
Block diagram of a memoryless digital modulation scheme.

$1 \le m \le 2^k$, signals regardless of the previously transmitted signals. This modulation scheme is equivalent to a mapping from $M = 2^k$ messages to M possible signals, as shown in Figure 3.1–1.

In a *modulation scheme with memory*, the mapping is from the set of the current k bits and the past $(L-1)k$ bits to the set of possible $M = 2^k$ messages. In this case the transmitted signal depends on the current k bits as well as the most recent $L-1$ blocks of k bits. This defines a finite-state machine with $2^{(L-1)k}$ states. The mapping that defines the modulation scheme can be viewed as a mapping from the current state and the current input of the modulator to the set of output signals resulting in a new state of the modulator. If at time instant $\ell - 1$ the modulator is in state $S_{\ell-1} \in \{1, 2, \ldots, 2^{(L-1)k}\}$ and the input sequence is $I_\ell \in \{1, 2, \ldots, 2^k\}$, then the modulator transmits the output $s_{m_\ell}(t)$ and moves to new state S_ℓ according to mappings

$$m_\ell = f_m(S_{\ell-1}, I_\ell) \tag{3.1–1}$$

$$S_\ell = f_s(S_{\ell-1}, I_\ell) \tag{3.1–2}$$

Parameters k and L and functions $f_m(\cdot,\cdot)$ and $f_s(\cdot,\cdot)$ completely describe the modulation scheme with memory. Parameter L is called the *constraint length* of modulation. The case of $L = 1$ corresponds to a memoryless modulation scheme.

Note the similarity between Equations 3.1–1 and 3.1–2 on one hand and Equations 2.7–43 and 2.7–44 on the other hand. Equation 3.1–2 represents the internal dynamics of a Markov chain where the future state depends on the current state and the input I_ℓ (which is a random variable), and Equation 3.1–1 states that the output m_ℓ depends on the state through random variable I_ℓ. Therefore, we can conclude that modulation systems with memory are effectively represented by Markov chains.

In addition to classifying the modulation as either memoryless or having memory, we may classify it as either linear or nonlinear. Linearity of a modulation method requires that the principle of superposition apply in the mapping of the digital sequence into successive waveforms. In nonlinear modulation, the superposition principle does not apply to signals transmitted in successive time intervals. We shall begin by describing memoryless modulation methods.

As indicated above, the modulator in a digital communication system maps a sequence of k binary symbols—which in case of equiprobable symbols carries k bits of information—into a set of corresponding signal waveforms $s_m(t)$, $1 \le m \le M$, where $M = 2^k$. We assume that these signals are transmitted at every T_s seconds, where T_s is called the *signaling interval*. This means that in each second

$$R_s = \frac{1}{T_s} \tag{3.1–3}$$

symbols are transmitted. Parameter R_s is called the *signaling rate* or *symbol rate*. Since each signal carries k bits of information, the *bit interval* T_b, i.e., the interval in which 1 bit of information is transmitted, is given by

$$T_b = \frac{T_s}{k} = \frac{T}{\log_2 M} \tag{3.1–4}$$

and the *bit rate* R is given by

$$R = kR_s = R_s \log_2 M \tag{3.1–5}$$

If the energy content of $s_m(t)$ is denoted by $\mathcal{E}_m$, then the *average signal energy* is given by

$$\mathcal{E}_{\text{avg}} = \sum_{m=1}^{M} p_m \mathcal{E}_m \tag{3.1–6}$$

where p_m indicates the probability of the mth signal (message probability). In the case of equiprobable messages, $p_m = 1/M$, and therefore,

$$\mathcal{E}_{\text{avg}} = \frac{1}{M} \sum_{m=1}^{M} \mathcal{E}_m \tag{3.1–7}$$

Obviously, if all signals have the same energy, then $\mathcal{E}_m = \mathcal{E}$ and $\mathcal{E}_{\text{avg}} = \mathcal{E}$. The average energy for transmission of 1 bit of information, or *average energy per bit*, when the signals are equiprobable is given by

$$\mathcal{E}_{\text{bavg}} = \frac{\mathcal{E}_{\text{avg}}}{k} = \frac{\mathcal{E}_{\text{avg}}}{\log_2 M} \tag{3.1–8}$$

If all signals have equal energy of $\mathcal{E}$, then

$$\mathcal{E}_b = \frac{\mathcal{E}}{k} = \frac{\mathcal{E}}{\log_2 M} \tag{3.1–9}$$

If a communication system is transmitting an average energy of $\mathcal{E}_{\text{bavg}}$ per bit, and it takes T_b seconds to transmit this average energy, then the average power sent by the transmitter is

$$P_{\text{avg}} = \frac{\mathcal{E}_{\text{bavg}}}{T_b} = R\mathcal{E}_{\text{bavg}} \tag{3.1–10}$$

which for the case of equal energy signals becomes

$$P = R\mathcal{E}_b \tag{3.1–11}$$

3.2

MEMORYLESS MODULATION METHODS

The waveforms $s_m(t)$ used to transmit information over the communication channel can be, in general, of any form. However, usually these waveforms are bandpass signals which may differ in amplitude or phase or frequency, or some combination of two or more signal parameters. We consider each of these signal types separately, beginning with digital *pulse amplitude modulation* (PAM). In all cases, we assume that the sequence of binary digits at the input to the modulator occurs at a rate of R bits/s.

3.2–1 Pulse Amplitude Modulation (PAM)

In digital PAM, the signal waveforms may be represented as

$$s_m(t) = A_m p(t), \quad 1 \le m \le M \tag{3.2–1}$$

where $p(t)$ is a pulse of duration T and $\{A_m, 1 \le m \le M\}$ denotes the set of M possible amplitudes corresponding to $M = 2^k$ possible k-bit blocks of symbols. Usually, the signal amplitudes A_m take the discrete values

$$A_m = 2m - 1 - M, \qquad m = 1, 2, \ldots, M \tag{3.2–2}$$

i.e., the amplitudes are $\pm 1, \pm 3, \pm 5, \ldots, \pm(M-1)$. The waveform $p(t)$ is a real-valued signal pulse whose shape influences the spectrum of the transmitted signal, as we shall observe later.

The energy in signal $s_m(t)$ is given by

$$\mathcal{E}_m = \int_{-\infty}^{\infty} A_m^2 p^2(t)\, dt \tag{3.2–3}$$

$$= A_m^2 \mathcal{E}_p \tag{3.2–4}$$

where $\mathcal{E}_p$ is the energy in $p(t)$. From this,

$$\begin{aligned}
\mathcal{E}_{\text{avg}} &= \frac{\mathcal{E}_p}{M} \sum_{m=1}^{M} A_m^2 \\
&= \frac{2\mathcal{E}_p}{M} \left(1^2 + 3^2 + 5^2 + \cdots + (M-1)^2\right) \\
&= \frac{2\mathcal{E}_p}{M} \times \frac{M(M^2-1)}{6} \\
&= \frac{(M^2-1)\mathcal{E}_p}{3}
\end{aligned} \tag{3.2–5}$$

and

$$\mathcal{E}_{\text{bavg}} = \frac{(M^2-1)\mathcal{E}_p}{3 \log_2 M} \tag{3.2–6}$$

What we described above is the baseband PAM in which no carrier modulation is present. In many cases the PAM signals are carrier-modulated bandpass signals with lowpass equivalents of the form $A_m g(t)$, where A_m and $g(t)$ are real. In this case

$$s_m(t) = \text{Re}\left[s_{ml}(t) e^{j2\pi f_c t}\right] \tag{3.2–7}$$

$$= \text{Re}\left[A_m g(t) e^{j2\pi f_c t}\right] = A_m g(t) \cos(2\pi f_c t) \tag{3.2–8}$$

where f_c is the carrier frequency. Comparing Equations 3.2–1 and 3.2–8, we note that if in the generic form of PAM signaling we substitute

$$p(t) = g(t)\cos(2\pi f_c t) \tag{3.2–9}$$

then we obtain the bandpass PAM. Using Equation 2.1–21, for bandpass PAM we have

$$\mathcal{E}_m = \frac{A_m^2}{2}\mathcal{E}_g \tag{3.2–10}$$

and from Equations 3.2–5 and 3.2–6 we conclude

$$\mathcal{E}_{\text{avg}} = \frac{(M^2-1)\mathcal{E}_g}{6} \tag{3.2–11}$$

and

$$\mathcal{E}_{\text{bavg}} = \frac{(M^2-1)\mathcal{E}_g}{6\log_2 M} \tag{3.2–12}$$

Clearly, PAM signals are one-dimensional ($N = 1$) since all are multiples of the same basic signals. Using the result of Example 2.2–6, we get

$$\phi(t) = \frac{p(t)}{\sqrt{\mathcal{E}_p}} \tag{3.2–13}$$

as the basis for the general PAM signal of the form $s_m(t) = A_m p(t)$ and

$$\phi(t) = \sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\cos 2\pi f_c t \tag{3.2–14}$$

as the basis for the bandpass PAM signal given in Equation 3.2–8. Using these basis signals, we have

$$\begin{aligned} s_m(t) &= A_m\sqrt{\mathcal{E}_p}\,\phi(t) && \text{for baseband PAM} \\ s_m(t) &= A_m\sqrt{\frac{\mathcal{E}_g}{2}}\,\phi(t) && \text{for bandpass PAM} \end{aligned} \tag{3.2–15}$$

From above the one-dimensional vector representations for these signals are of the form

$$s_m = A_m\sqrt{\mathcal{E}_p}, \qquad A_m = \pm1, \pm3, \ldots, \pm(M-1) \tag{3.2–16}$$

$$s_m = A_m\sqrt{\frac{\mathcal{E}_g}{2}}, \qquad A_m = \pm1, \pm3, \ldots, \pm(M-1) \tag{3.2–17}$$

The corresponding signal space diagrams for $M = 2$, $M = 4$, and $M = 8$ are shown in Figure 3.2–1.

The bandpass digital PAM is also called *amplitude-shift keying* (ASK). The mapping or assignment of k information bits to the $M = 2^k$ possible signal amplitudes may be done in a number of ways. The preferred assignment is one in which the adjacent

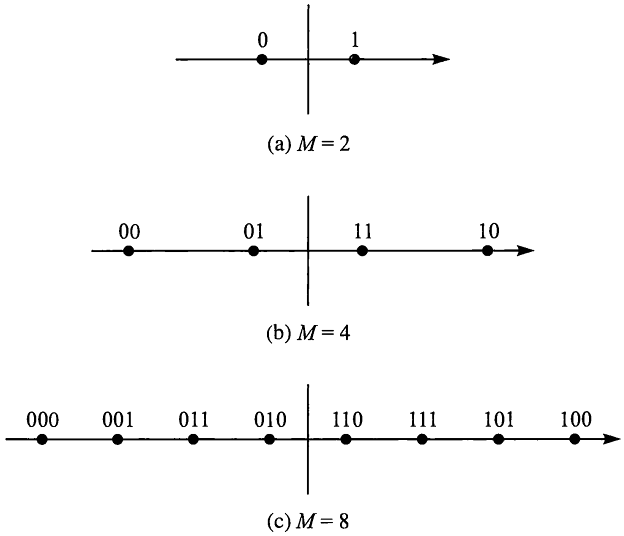


FIGURE 3.2–1
Constellation for PAM signaling.

signal amplitudes differ by one binary digit as illustrated in Figure 3.2–1. This mapping is called *Gray coding*. It is important in the demodulation of the signal because the most likely errors caused by noise involve the erroneous selection of an adjacent amplitude to the transmitted signal amplitude. In such a case, only a single bit error occurs in the k-bit sequence.

We note that the Euclidean distance between any pair of signal points is

$$\begin{aligned} d_{mn} &= \sqrt{\|s_m - s_n\|^2} && (3.2\text{–}18)\\ &= |A_m - A_n|\sqrt{\mathcal{E}_p} && (3.2\text{–}19)\\ &= |A_m - A_n|\sqrt{\frac{\mathcal{E}_g}{2}} && (3.2\text{–}20)\end{aligned}$$

where the last relation corresponds to a bandpass PAM. For adjacent signal points $|A_m - A_n| = 2$, and hence the minimum distance of the constellation is given by

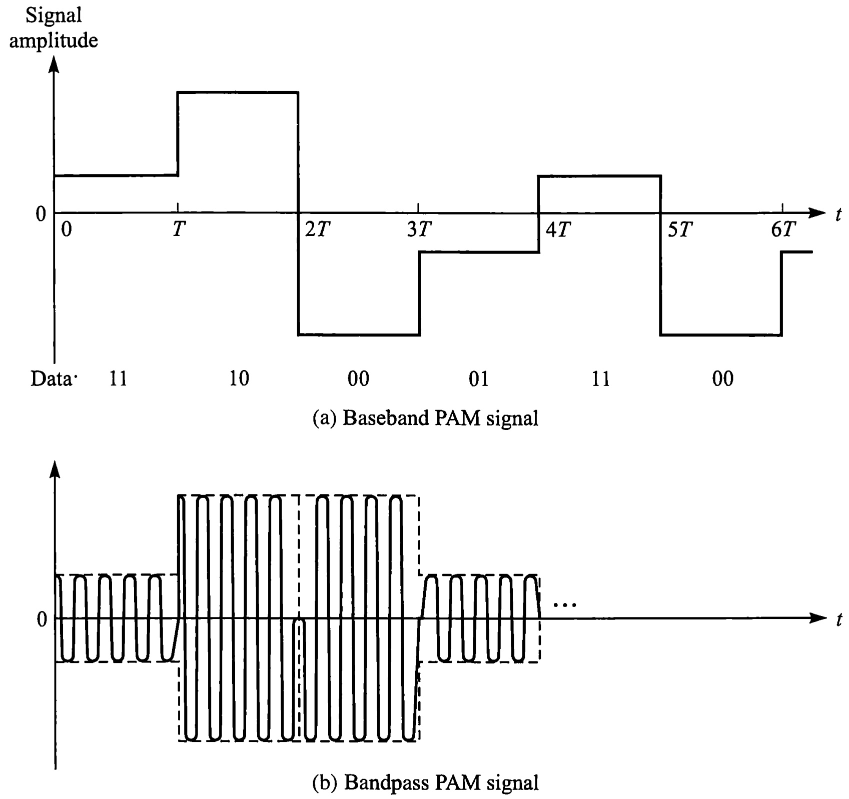
$$d_{\min} = 2\sqrt{\mathcal{E}_p} = \sqrt{2\mathcal{E}_g} \qquad (3.2\text{–}21)$$

We can express the minimum distance of an M-ary PAM system in terms of its $\mathcal{E}_{\text{bavg}}$ by solving Equations 3.2–6 and 3.2–12 for $\mathcal{E}_p$ and $\mathcal{E}_g$, respectively, and substituting the result in Equation 3.2–21. The resulting expression is

$$d_{\min} = \sqrt{\frac{12\log_2 M}{M^2 - 1}\mathcal{E}_{\text{bavg}}} \qquad (3.2\text{–}22)$$

The carrier-modulated PAM signal represented by Equation 3.2–8 is a double-sideband (DSB) signal and requires twice the channel bandwidth of the equivalent lowpass signal for transmission. Alternatively, we may use single-sideband (SSB) PAM, which has the representation (lower or upper sideband)

$$s_m(t) = \text{Re}\left[A_m\left(g(t) \pm j\hat{g}(t)\right)e^{j2\pi f_c t}\right], \qquad m = 1, 2, \ldots, M \qquad (3.2\text{–}23)$$



(a) Baseband PAM signal

(b) Bandpass PAM signal

FIGURE 3.2–2
Example of (a) baseband and (b) carrier-modulated PAM signals.

where $\hat{g}(t)$ is the Hilbert transform of $g(t)$. Thus, the bandwidth of the SSB signal is one-half that of the DSB signal.

A four-amplitude level baseband PAM signal is illustrated in Figure 3.2–2(a). The carrier-modulated version of the signal is shown in Figure 3.2–2(b).

In the special case of $M = 2$, or binary signals, the PAM waveforms have the special property that $s_1(t) = -s_2(t)$. Hence, these two signals have the same energy and a cross-correlation coefficient of -1. Such signals are called *antipodal*. This case is sometimes called *binary antipodal signaling*.

3.2–2 Phase Modulation

In digital phase modulation, the M signal waveforms are represented as

$$\begin{aligned} s_m(t) &= \text{Re}\left[g(t)e^{j\frac{2\pi(m-1)}{M}}e^{j2\pi f_c t}\right], \quad m = 1, 2, \ldots, M \\ &= g(t)\cos\left[2\pi f_c t + \frac{2\pi}{M}(m-1)\right] \\ &= g(t)\cos\left(\frac{2\pi}{M}(m-1)\right)\cos 2\pi f_c t - g(t)\sin\left(\frac{2\pi}{M}(m-1)\right)\sin 2\pi f_c t \end{aligned} \tag{3.2–24}$$

where $g(t)$ is the signal pulse shape and $\theta_m = 2\pi(m-1)/M$, $m = 1, 2, \ldots, M$, is the M possible phases of the carrier that convey the transmitted information. Digital phase modulation is usually called phase-shift keying (PSK). We note that these signal waveforms have equal energy. From Equation 2.1–21,

$$\mathcal{E}_{\text{avg}} = \mathcal{E}_m = \frac{1}{2}\mathcal{E}_g \tag{3.2–25}$$

and therefore,

$$\mathcal{E}_{\text{bavg}} = \frac{\mathcal{E}_g}{2\log_2 M} \tag{3.2–26}$$

For this case, instead of $\mathcal{E}_{\text{avg}}$ and $\mathcal{E}_{\text{bavg}}$ we use the notation $\mathcal{E}$ and $\mathcal{E}_b$.

Using the result of Example 2.1–1, we note that $g(t)\cos 2\pi f_c T$ and $g(t)\sin 2\pi f_c t$ are orthogonal, and therefore $\phi_1(t)$ and $\phi_2(t)$ given as

$$\phi_1(t) = \sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\cos 2\pi f_c t \tag{3.2–27}$$

$$\phi_2(t) = -\sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\sin 2\pi f_c t \tag{3.2–28}$$

can be used for expansion of $s_m(t)$, $1 \le m \le M$, as

$$s_m(t) = \sqrt{\frac{\mathcal{E}_g}{2}}\cos\left(\frac{2\pi}{M}(m-1)\right)\phi_1(t) + \sqrt{\frac{\mathcal{E}_g}{2}}\sin\left(\frac{2\pi}{M}(m-1)\right)\phi_2(t) \tag{3.2–29}$$

therefore the signal space dimensionality is $N = 2$ and the resulting vector representations are

$$s_m = \left(\sqrt{\frac{\mathcal{E}_g}{2}}\cos\left(\frac{2\pi}{M}(m-1)\right), \sqrt{\frac{\mathcal{E}_g}{2}}\sin\left(\frac{2\pi}{M}(m-1)\right)\right), \quad m = 1, 2, \ldots, M \tag{3.2–30}$$

Signal space diagrams for BPSK (binary PSK, $M = 2$), QPSK (quaternary PSK, $M = 4$), and 8-PSK are shown in Figure 3.2–3. We note that BPSK corresponds to one-dimensional signals, which are identical to binary PAM signals. These signaling schemes are special cases of binary antipodal signaling discussed earlier.

As is the case with PAM, the mapping or assignment of k information bits to the $M = 2^k$ possible phases may be done in a number of ways. The preferred assignment is Gray encoding, so that the most likely errors caused by noise will result in a single bit error in the k-bit symbol.

The Euclidean distance between signal points is

$$\begin{aligned} d_{mn} &= \sqrt{\|s_m - s_n\|^2} \\ &= \sqrt{\mathcal{E}_g\left[1 - \cos\left(\frac{2\pi}{M}(m-n)\right)\right]} \end{aligned} \tag{3.2–31}$$

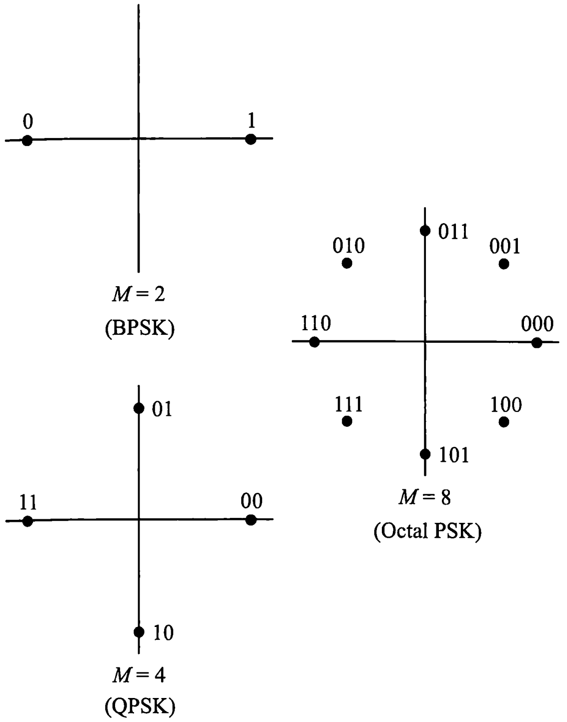


FIGURE 3.2–3
Signal space diagrams for BPSK, QPSK, and 8-PSK.

and the minimum distance corresponding to $|m - n| = 1$ is

$$d_{\min} = \sqrt{\mathcal{E}_g \left(1 - \cos \frac{2\pi}{M}\right)} = \sqrt{2\mathcal{E}_g \sin^2 \frac{\pi}{M}} \tag{3.2–32}$$

Solving Equation 3.2–26 for $\mathcal{E}_g$ and substituting the result in Equation 3.2–32 result in

$$d_{\min} = 2\sqrt{\left(\log_2 M \times \sin^2 \frac{\pi}{M}\right) \mathcal{E}_b} \tag{3.2–33}$$

For large values of M, we have $\sin \frac{\pi}{M} \approx \frac{\pi}{M}$, and $d_{\min}$ can be approximated by

$$d_{\min} \approx 2\sqrt{\frac{\pi^2 \log_2 M}{M^2} \mathcal{E}_b} \tag{3.2–34}$$

A variant of four-phase PSK (QPSK), called $\frac{\pi}{4}$-QPSK, is obtained by introducing an additional $\pi/4$ phase shift in the carrier phase in each symbol interval. This phase shift facilitates symbol synchronization.

3.2–3 Quadrature Amplitude Modulation

The bandwidth efficiency of PAM/SSB can also be obtained by simultaneously impressing two separate k-bit symbols from the information sequence on two quadrature carriers $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$. The resulting modulation technique is called quadrature PAM or QAM, and the corresponding signal waveforms may be expressed as

$$\begin{aligned} s_m(t) &= \operatorname{Re}\left[(A_{mi} + jA_{mq})g(t)e^{j2\pi f_c t}\right] \\ &= A_{mi}g(t)\cos 2\pi f_c t - A_{mq}g(t)\sin 2\pi f_c t, \qquad m = 1, 2, \ldots, M \end{aligned} \tag{3.2–35}$$

where A_{mi} and A_{mq} are the information-bearing signal amplitudes of the quadrature carriers and $g(t)$ is the signal pulse. Alternatively, the QAM signal waveforms may be expressed as

$$\begin{aligned} s_m(t) &= \operatorname{Re}\left[r_m e^{j\theta_m} e^{j2\pi f_c t}\right] \\ &= r_m \cos(2\pi f_c t + \theta_m) \end{aligned} \tag{3.2–36}$$

where $r_m = \sqrt{A_{mi}^2 + A_{mq}^2}$ and $\theta_m = \tan^{-1}(A_{mq}/A_{mi})$. From this expression, it is apparent that the QAM signal waveforms may be viewed as combined amplitude (r_m) and phase (θ_m) modulation. In fact, we may select any combination of M_1-level PAM and M_2-phase PSK to construct an $M = M_1M_2$ combined PAM-PSK signal constellation. If $M_1 = 2^n$ and $M_2 = 2^m$, the combined PAM-PSK signal constellation results in the simultaneous transmission of $m + n = \log_2 M_1M_2$ binary digits occurring at a symbol rate $R/(m + n)$.

From Equation 3.2–35, it can be seen that, similar to the PSK case, $\phi_1(t)$ and $\phi_2(t)$ given in Equations 3.2–27 and 3.2–28 can be used as an orthonormal basis for expansion of QAM signals. The dimensionality of the signal space for QAM is $N = 2$. Using this basis, we have

$$s_m(t) = A_{mi}\sqrt{\frac{\mathcal{E}_g}{2}}\phi_1(t) + A_{mq}\sqrt{\frac{\mathcal{E}_g}{2}}\phi_2(t) \tag{3.2–37}$$

which results in vector representations of the form

$$\begin{aligned} \boldsymbol{s}_m &= (s_{m1}, s_{m2}) \\ &= \left(A_{mi}\sqrt{\frac{\mathcal{E}_g}{2}}, A_{mq}\sqrt{\frac{\mathcal{E}_g}{2}}\right) \end{aligned} \tag{3.2–38}$$

and

$$\mathcal{E}_m = \|\boldsymbol{s}_m\|^2 = \frac{\mathcal{E}_g}{2}\left(A_{mi}^2 + A_{mq}^2\right) \tag{3.2–39}$$

Examples of signal space diagrams for combined PAM-PSK are shown in Figure 3.2–4, for $M = 8$ and $M = 16$.

The Euclidean distance between any pair of signal vectors in QAM is

$$\begin{aligned} d_{mn} &= \sqrt{\|\boldsymbol{s}_m - \boldsymbol{s}_n\|^2} \\ &= \sqrt{\frac{\mathcal{E}_g}{2}\left[(A_{mi} - A_{ni})^2 + (A_{mq} - A_{nq})^2\right]} \end{aligned} \tag{3.2–40}$$

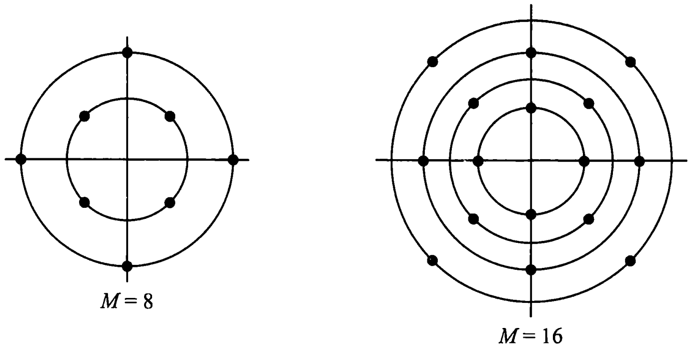


FIGURE 3.2–4
Examples of combined PAM-PSK constellations.

In the special case where the signal amplitudes take the set of discrete values $\{(2m-1-M), m=1,2,\ldots,M\}$, the signal space diagram is rectangular, as shown in Figure 3.2–5. In this case, the Euclidean distance between adjacent points, i.e., the minimum distance, is

$$d_{\min} = \sqrt{2\mathcal{E}_g} \tag{3.2–41}$$

which is the same result as for PAM. In the special case of a rectangular constellation with $M = 2^{2k_1}$, i.e., $M = 4, 16, 64, 256, \ldots$, and with amplitudes of $\pm1, \pm3, \ldots, \pm(\sqrt{M}-1)$ on both directions, from Equation 3.2–39 we have

$$\begin{aligned} \mathcal{E}_{\text{avg}} &= \frac{1}{M}\frac{\mathcal{E}_g}{2}\sum_{m=1}^{\sqrt{M}}\sum_{n=1}^{\sqrt{M}}(A_m^2 + A_n^2) \\ &= \frac{\mathcal{E}_g}{2M} \times \frac{2M(M-1)}{3} \\ &= \frac{M-1}{3}\mathcal{E}_g \end{aligned} \tag{3.2–42}$$

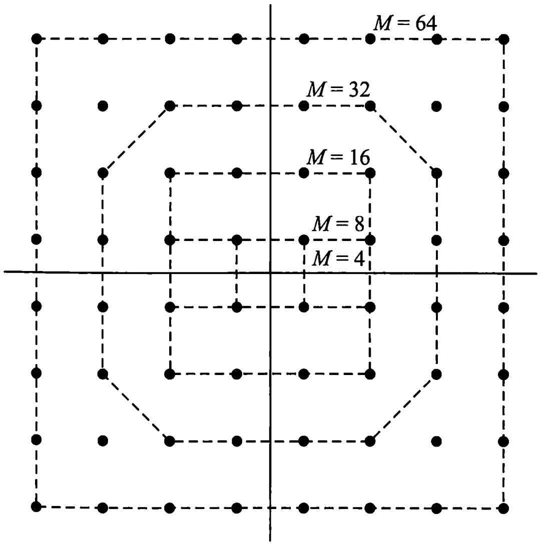


FIGURE 3.2–5
Several signal space diagrams for rectangular QAM.

from which

$$\mathcal{E}_{\text{bavg}} = \frac{M-1}{3\log_2 M}\mathcal{E}_g \tag{3.2–43}$$

Using Equation 3.2–41, we obtain

$$d_{\min} = \sqrt{\frac{6\log_2 M}{M-1}\mathcal{E}_{\text{bavg}}} \tag{3.2–44}$$

Table 3.2–1 summarizes some basic properties of the modulation schemes discussed above. In this table it is assumed that for PAM and QAM signaling, the amplitudes are $\pm 1, \pm 3, \ldots, \pm(M-1)$ and the QAM signaling has a rectangular $\sqrt{M} \times \sqrt{M}$ constellation.

From the discussion of bandpass PAM, PSK, and QAM, it is clear that all these signaling schemes are of the general form

$$s_m(t) = \text{Re}\left[A_m g(t) e^{j2\pi f_c t}\right], \quad m = 1, 2, \ldots, M \tag{3.2–45}$$

where A_m is determined by the signaling scheme. For PAM, A_m is real, generally equal to $\pm 1, \pm 3, \ldots, \pm(M-1)$, for M-ary PSK, A_m is complex and equal to $e^{j\frac{2\pi}{M}(m-1)}$; and finally for QAM, A_m is a general complex number $A_m = A_{mi} + jA_{mq}$. In this sense it is seen that these three signaling schemes belong to the same family, and PAM and PSK can be considered as special cases of QAM. In QAM signaling, both amplitude and phase carry information, whereas in PAM and PSK only amplitude or phase carries the information. Also note that in these schemes the dimensionality of the signal space is rather low (one for PAM and two for PSK and QAM) and is independent of the constellation size M. The structure of the modulator for this general class of signaling schemes is shown in Figure 3.2–6, where $\phi_1(t)$ and $\phi_2(t)$ are given by Equation 3.2–27. Note that the modulator consists of a vector mapper, which maps each of the M messages onto a constellation of size M, followed by a two-dimensional (or one-dimensional, in case of PAM) vector to signal mapper as was previously shown in Figure 2.2–2.

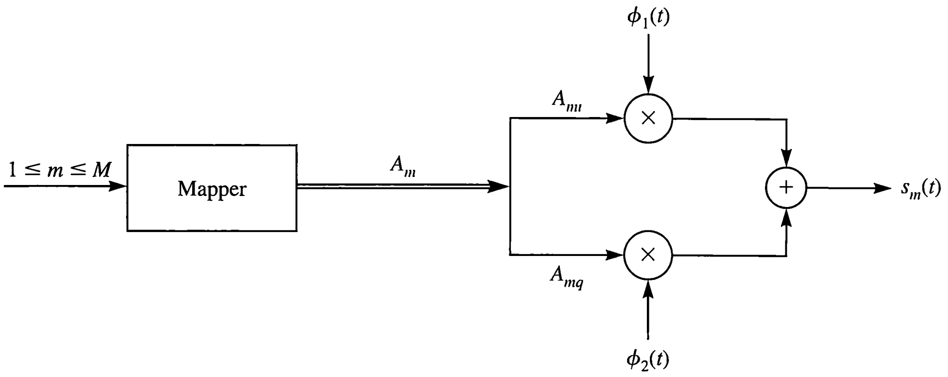


FIGURE 3.2–6
A general QAM modulator.

TABLE 3.2–1

Comparison of PAM, PSK, and QAM

Signaling Scheme	$s_m(t)$	s_m	$E_{\mathbf{avg}}$	$E_{\mathbf{bavg}}$	$d_{\min}$
Baseband PAM	$A_m p(t)$	$A_m \sqrt{\mathcal{E}_p}$	$\frac{2(M^2-1)}{3} \mathcal{E}_p$	$\frac{2(M^2-1)}{3 \log_2 M} \mathcal{E}_p$	$\sqrt{\frac{6 \log_2 M}{M^2-1} \mathcal{E}_{\text{bavg}}}$
Bandpass PAM	$A_m g(t) \cos 2\pi f_c t$	$A_m \sqrt{\frac{\mathcal{E}_g}{2}}$	$\frac{M^2-1}{3} \mathcal{E}_g$	$\frac{M^2-1}{3 \log_2 M} \mathcal{E}_g$	$\sqrt{\frac{6 \log_2 M}{M^2-1} \mathcal{E}_{\text{bavg}}}$
PSK	$g(t) \cos \left[2\pi f_c t + \frac{2\pi}{M}(m-1)\right]$	$\sqrt{\frac{\mathcal{E}_g}{2}} \left(\cos \frac{2\pi}{M}(m-1), \sin \frac{2\pi}{M}(m-1)\right)$	$\frac{1}{2} \mathcal{E}_g$	$\frac{1}{2 \log_2 M} \mathcal{E}_g$	$2 \sqrt{\log_2 M \ \sin^2\left(\frac{\pi}{M}\right) \mathcal{E}_{\text{bavg}}}$
QAM	$A_{mi} g(t) \cos 2\pi f_c t - A_{mq} g(t) \sin 2\pi f_c t$	$\sqrt{\frac{\mathcal{E}_g}{2}} (A_{mi}, A_{mq})$	$\frac{M-1}{3} \mathcal{E}_g$	$\frac{M-1}{3 \log_2 M} \mathcal{E}_g$	$\sqrt{\frac{6 \log_2 M}{M-1} \mathcal{E}_{\text{bavg}}}$

3.2–4 Multidimensional Signaling

It is apparent from the discussion above that the digital modulation of the carrier amplitude and phase allows us to construct signal waveforms that correspond to two-dimensional vectors and signal space diagrams. If we wish to construct signal waveforms corresponding to higher-dimensional vectors, we may use either the time domain or the frequency domain or both to increase the number of dimensions. Suppose we have N-dimensional signal vectors. For any N, we may subdivide a time interval of length $T_1 = NT$ into N subintervals of length $T = T_1/N$. In each subinterval of length T, we may use binary PAM (a one-dimensional signal) to transmit an element of the N-dimensional signal vector. Thus, the N time slots are used to transmit the N-dimensional signal vector. If N is even, a time slot of length T may be used to simultaneously transmit two components of the N-dimensional vector by modulating the amplitude of quadrature carriers independently by the corresponding components. In this manner, the N-dimensional signal vector is transmitted in $\frac{1}{2}NT$ seconds ($\frac{1}{2}N$ time slots). Alternatively, a frequency band of width $N\Delta f$ may be subdivided into N frequency slots each of width Δf. An N-dimensional signal vector can be transmitted over the channel by simultaneously modulating the amplitude of N carriers, one in each of the N frequency slots. Care must be taken to provide sufficient frequency separation Δf between successive carriers so that there is no cross-talk interference among the signals on the N carriers. If quadrature carriers are used in each frequency slot, the N-dimensional vector (even N) may be transmitted in $\frac{1}{2}N$ frequency slots, thus reducing the channel bandwidth utilization by a factor of 2. More generally, we may use both the time and frequency domains jointly to transmit an N-dimensional signal vector. For example, Figure 3.2–7 illustrates a subdivision of the time and frequency axes into 12 slots. Thus, an $N = 12$-dimensional signal vector may be transmitted by PAM or an $N = 24$-dimensional signal vector may be transmitted by use of two quadrature carriers (QAM) in each slot.

Orthogonal Signaling

Orthogonal signals are defined as a set of equal energy signals $s_m(t)$, $1 \le m \le M$, such that

$$\langle s_m(t), s_n(t) \rangle = 0, \qquad m \neq n \text{ and } 1 \le m, n \le M \qquad (3.2\text{–}46)$$

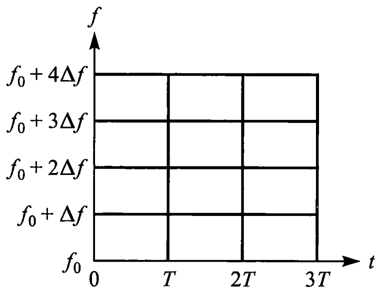


FIGURE 3.2–7
Subdivision of time and frequency axes into distinct slots.

With this definition it is clear that

$$\langle s_m(t), s_n(t)\rangle = \begin{cases} \mathcal{E} & m = n \\ 0 & m \neq n \end{cases} \qquad 1 \leq m, n \leq M \tag{3.2–47}$$

Obviously the signals are linearly independent and hence $N = M$. The orthonormal set $\{\phi_j(t),\ 1 \leq i \leq N\}$ given by

$$\phi_j(t) = \frac{s_j(t)}{\sqrt{\mathcal{E}}}, \qquad 1 \leq j \leq N \tag{3.2–48}$$

can be used as an orthonormal basis for representation of $\{s_m(t),\ 1 \leq m \leq M\}$. The resulting vector representation of the signals will be

$$\begin{aligned} s_1 &= (\sqrt{\mathcal{E}}, 0, 0, \ldots, 0) \\ s_2 &= (0, \sqrt{\mathcal{E}}, 0, \ldots, 0) \\ \vdots &= \vdots \\ s_M &= (0, 0, \ldots, 0, \sqrt{\mathcal{E}}) \end{aligned} \tag{3.2–49}$$

From Equation 3.2–49 it is seen that for all $m \neq n$ we have

$$d_{mn} = \sqrt{2\mathcal{E}} \tag{3.2–50}$$

and therefore,

$$d_{\min} = \sqrt{2\mathcal{E}} \tag{3.2–51}$$

in all orthogonal signaling schemes. Using the relation

$$\mathcal{E}_b = \frac{\mathcal{E}}{\log_2 M} \tag{3.2–52}$$

we conclude that

$$d_{\min} = \sqrt{2 \log_2 M\ \mathcal{E}_b} \tag{3.2–53}$$

Frequency-Shift Keying (FSK) As a special case of the construction of orthogonal signals, let us consider the construction of orthogonal signal waveforms that differ in frequency and are represented as

$$\begin{aligned} s_m(t) &= \operatorname{Re}\left[s_{ml}(t) e^{j2\pi f_c t}\right], \qquad 1 \leq m \leq M, \quad 0 \leq t \leq T \\ &= \sqrt{\frac{2\mathcal{E}}{T}} \cos\left(2\pi f_c t + 2\pi m\ \Delta f\ t\right) \end{aligned} \tag{3.2–54}$$

where

$$s_{ml}(t) = \sqrt{\frac{2\mathcal{E}}{T}} e^{j2\pi m \Delta f t}, \qquad 1 \leq m \leq M, \quad 0 \leq t \leq T \tag{3.2–55}$$

The coefficient $\sqrt{\frac{2\mathcal{E}}{T}}$ is introduced to guarantee that each signal has an energy equal to $\mathcal{E}$. This type of signaling, in which the messages are transmitted by signals that differ in frequency, is called *frequency-shift keying* (FSK). Note a major difference between FSK and QAM signals (of which ASK and PSK can be considered as special cases). In QAM signaling the lowpass equivalent of the signal is of the form $A_m g(t)$ where A_m is a complex number. Therefore the sum of two lowpass equivalent signals corresponding to two different signals is of the general form of the lowpass equivalent of a QAM signal. In this sense, the sum of two QAM signals is another QAM signal. For this reason, ASK, PSK, and QAM are sometimes called *linear modulation schemes*. On the other hand, FSK signaling does not satisfy this property, and therefore it belongs to the class of *nonlinear modulation schemes*.

By using Equation 2.1–26, it is clear that for this set of signals to be orthogonal, we need to have

$$\operatorname{Re}\left[\int_0^T s_{ml}(t) s_{nl}(t)\, dt\right] = 0 \tag{3.2–56}$$

for all $m \neq n$. But

$$\begin{aligned}\langle s_{ml}(t), s_{nl}(t)\rangle &= \frac{2\mathcal{E}}{T}\int_0^T e^{j2\pi(m-n)\Delta f t}\, dt \\ &= \frac{2\mathcal{E}\sin(\pi T(m-n)\Delta f)}{\pi T(m-n)\Delta f}\, e^{j\pi T(m-n)\Delta f}\end{aligned} \tag{3.2–57}$$

and

$$\begin{aligned}\operatorname{Re}\left[\langle s_{ml}(t), s_{nl}(t)\rangle\right] &= \frac{2\mathcal{E}\sin(\pi T(m-n)\Delta f)}{\pi T(m-n)\Delta f}\cos(\pi T(m-n)\Delta f) \\ &= \frac{2\mathcal{E}\sin(2\pi T(m-n)\Delta f)}{2\pi T(m-n)\Delta f} \\ &= 2\mathcal{E}\operatorname{sinc}(2T(m-n)\Delta f)\end{aligned} \tag{3.2–58}$$

From Equation 3.2–58 we observe that $s_m(t)$ and $s_n(t)$ are orthogonal for all $m \neq n$ if and only if $\operatorname{sinc}(2T(m-n)\Delta f) = 0$ for all $m \neq n$. This is the case if $\Delta f = k/2T$ for some positive integer k. The minimum frequency separation Δf that guarantees orthogonality is $\Delta f = 1/2T$. Note that $\Delta f = \frac{1}{2T}$ is the minimum frequency separation that guarantees $\langle s_{ml}(t), s_{nl}(t)\rangle = 0$, thus guaranteeing the orthogonality of the baseband, as well as the bandpass, frequency-modulated signals.

Hadamard signals are orthogonal signals which are constructed from Hadamard matrices. Hadamard matrices $\boldsymbol{H}_n$ are $2^n \times 2^n$ matrices for $n = 1, 2, \ldots$ defined by the following recursive relation

$$\begin{aligned}\boldsymbol{H}_0 &= [1] \\ \boldsymbol{H}_{n+1} &= \begin{bmatrix} \boldsymbol{H}_n & \boldsymbol{H}_n \\ \boldsymbol{H}_n & -\boldsymbol{H}_n \end{bmatrix}\end{aligned} \tag{3.2–59}$$

With this definition we have

$$
\begin{aligned}
\boldsymbol{H}_1 &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\
\boldsymbol{H}_2 &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \\
\boldsymbol{H}_3 &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}
\end{aligned}
\tag{3.2–60}
$$

Hadamard matrices are symmetric matrices whose rows (and, by symmetry, columns) are orthogonal. Using these matrices, we can generate orthogonal signals. For instance, using $\boldsymbol{H}_2$ would result in the set of signals

$$
\begin{aligned}
s_1 &= [\sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}}] \\
s_2 &= [\sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}}] \\
s_3 &= [\sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}}] \\
s_4 &= [\sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}} \quad -\sqrt{\mathcal{E}} \quad \sqrt{\mathcal{E}}]
\end{aligned}
\tag{3.2–61}
$$

This set of orthogonal signals may be used to modulate any four-dimensional orthonormal basis $\{\phi_j(t)\}_{j=1}^4$ to generate signals of the form

$$
s_m(t) = \sum_{j=1}^{4} s_{mj}\phi_j(t), \qquad 1 \le m \le 4 \tag{3.2–62}
$$

Note that the energy in each signal is $4\mathcal{E}$, and each signal carries 2 bits of information, hence $\mathcal{E}_b = 2\mathcal{E}$.

Biorthogonal Signaling

A set of M biorthogonal signals can be constructed from $\frac{1}{2}M$ orthogonal signals by simply including the negatives of the orthogonal signals. Thus, we require $N = \frac{1}{2}M$ dimensions for the construction of a set of M biorthogonal signals. Figure 3.2–8 illustrates the biorthogonal signals for $M = 4$ and 6. We note that the correlation between any pair of waveforms is $\rho = -1$ or 0. The corresponding distances are $d = 2\sqrt{\mathcal{E}}$ or $\sqrt{2\mathcal{E}}$, with the latter being the minimum distance.

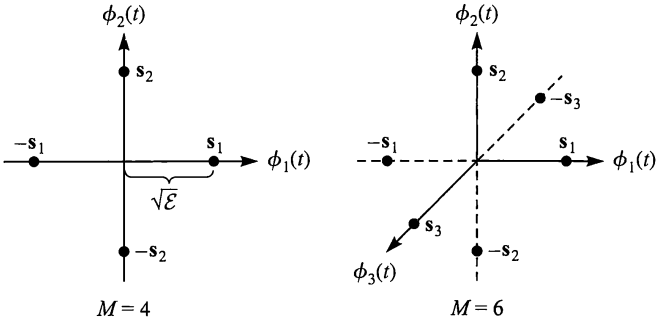


FIGURE 3.2–8
Signal space diagram for $M = 4$ and $M = 6$ biorthogonal signals.

Simplex Signaling

Suppose we have a set of M orthogonal waveforms $\{s_m(t)\}$ or, equivalently, their vector representation $\{\boldsymbol{s}_m\}$. Their mean is

$$\bar{\boldsymbol{s}} = \frac{1}{M}\sum_{m=1}^{M} \boldsymbol{s}_m \tag{3.2–63}$$

Now, let us construct another set of M signals by subtracting the mean from each of the M orthogonal signals. Thus,

$$\boldsymbol{s}'_m = \boldsymbol{s}_m - \bar{\boldsymbol{s}}, \qquad m = 1, 2, \ldots, M \tag{3.2–64}$$

The effect of the subtraction is to translate the origin of the m orthogonal signals to the point $\bar{\boldsymbol{s}}$. The resulting signal waveforms are called *simplex signals* and have the following properties. First, the energy per waveform is

$$\begin{aligned} \|\boldsymbol{s}'_m\|^2 &= \|\boldsymbol{s}_m - \bar{\boldsymbol{s}}\|^2 \\ &= \mathcal{E} - \frac{2}{M}\mathcal{E} + \frac{1}{M}\mathcal{E} \\ &= \mathcal{E}\left(1 - \frac{1}{M}\right) \end{aligned} \tag{3.2–65}$$

Second, the cross-correlation of any pair of signals is

$$\begin{aligned} \text{Re}\,[\rho_{mn}] &= \frac{\boldsymbol{s}'_m \cdot \boldsymbol{s}'_n}{\|\boldsymbol{s}'_m\|\,\|\boldsymbol{s}'_n\|} \\ &= \frac{-1/M}{1 - 1/M} = -\frac{1}{M-1} \end{aligned} \tag{3.2–66}$$

Hence, the set of simplex waveforms is equally correlated and requires less energy, by the factor $1 - 1/M$, than the set of orthogonal waveforms. Since only the origin was translated, the distance between any pair of signal points is maintained at $d = \sqrt{2\mathcal{E}}$, which is the same as the distance between any pair of orthogonal signals. Figure 3.2–9 illustrates the simplex signals for $M = 2$, 3, and 4. Note that the signal dimensionality is $N = M - 1$.

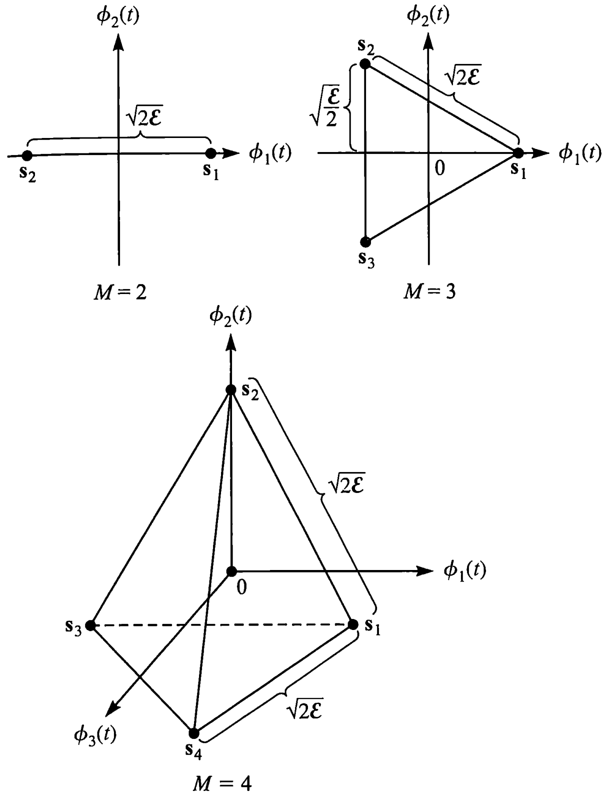

$\phi_2(t)$
$\sqrt{2\mathcal{E}}$
$\phi_1(t)$
s_2
s_1
$M=2$
$\phi_2(t)$
s_2
$\sqrt{2\mathcal{E}}$
$\sqrt{\frac{\mathcal{E}}{2}}$
0
s_1
$\phi_1(t)$
s_3
$M=3$
$\phi_2(t)$
s_2
$\sqrt{2\mathcal{E}}$
$\phi_1(t)$
0
s_3
s_1
$\sqrt{2\mathcal{E}}$
$\phi_3(t)$
s_4
$M=4$

FIGURE 3.2–9
Signal space diagrams for M-ary simplex signals.

Note that the class of orthogonal, biorthogonal, and simplex signals has many common properties. The signal space dimensionality in this class is highly dependent on the constellation size. This is in contrast to PAM, PSK, and QAM systems. Also, for fixed $\mathcal{E}_b$, the minimum distance $d_{\min}$ in these systems increases with increasing M. This again is in sharp contrast to PAM, PSK, and QAM signaling. We will see later in Chapter 4 that similar contrasts in power and bandwidth efficiency exist between these two classes of signaling schemes.

Signal Waveforms from Binary Codes

A set of M signaling waveforms can be generated from a set of M binary code words of the form

$$\mathbf{c}_m = [c_{m1}\, c_{m2} \cdots c_{mN}], \qquad m = 1, 2, \ldots, M \tag{3.2–67}$$

where $c_{mj} = 0$ or 1 for all m and j. Each component of a code word is mapped into an elementary binary PSK waveform as follows:

$$\begin{aligned} c_{mj} = 1 &\Longrightarrow \sqrt{\frac{2\mathcal{E}_c}{T_c}} \cos 2\pi f_c t, \qquad 0 \le t \le T_c \\ c_{mj} = 0 &\Longrightarrow -\sqrt{\frac{2\mathcal{E}_c}{T_c}} \cos 2\pi f_c t, \qquad 0 \le t \le T_c \end{aligned} \tag{3.2–68}$$

where $T_c = T/N$ and $\mathcal{E}_c = \mathcal{E}/N$. Thus, the M code words $\{\mathbf{c}_m\}$ are mapped into a set of M waveforms $\{s_m(t)\}$. The waveforms can be represented in vector form as

$$\mathbf{s}_m = [s_{m1}\, s_{m2} \cdots s_{mN}], \qquad m = 1, 2, \ldots, M \tag{3.2–69}$$

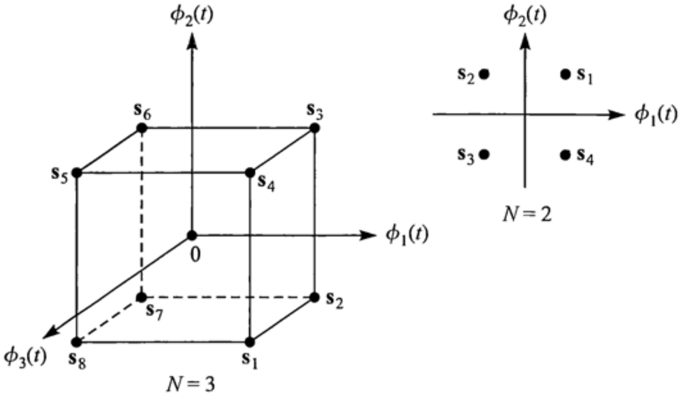


FIGURE 3.2–10
Signal space diagrams for signals generated from binary codes.

where $s_{mj} = \pm\sqrt{\mathcal{E}/N}$ for all m and j. Also N is called the block length of the code, and it is the dimension of the M waveforms. We note that there are 2^N possible waveforms that can be constructed from the 2^N possible binary code words. We may select a subset of $M < 2^N$ signal waveforms for transmission of the information. We also observe that the 2^N possible signal points correspond to the vertices of an N-dimensional hypercube with its center at the origin. Figure 3.2–10 illustrates the signal points in $N = 2$ and 3 dimensions. Each of the M waveforms has energy $\mathcal{E}$. The cross-correlation between any pair of waveforms depends on how we select the M waveforms from the 2^N possible waveforms. This topic is treated in detail in Chapters 7 and 8. Clearly, any adjacent signal points have a cross-correlation coefficient

$$\rho = \frac{\mathcal{E}(1 - 2/N)}{\mathcal{E}} = \frac{N-2}{N} \tag{3.2–70}$$

and a corresponding distance of

$$d_{\min} = \sqrt{2\mathcal{E}(1-\rho)} = \sqrt{4\mathcal{E}/N} \tag{3.2–71}$$

The Hadamard signals described previously are special cases of signals based on codes.

3.3
SIGNALING SCHEMES WITH MEMORY

We have seen before that signaling schemes with memory can be best explained in terms of Markov chains and finite-state machines. The state transition and the outputs of the Markov chain are governed by

$$\begin{aligned} m_\ell &= f_m(S_{\ell-1}, I_\ell) \\ S_\ell &= f_s(S_{\ell-1}, I_\ell) \end{aligned} \tag{3.3–1}$$

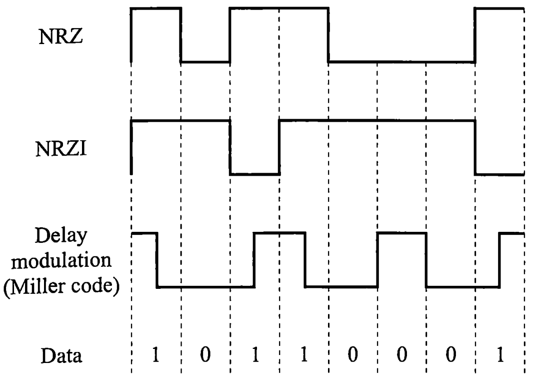


FIGURE 3.3–1
Examples of baseband signals.

where I_ℓ denotes the information sequence and m_ℓ is the index of the transmitted signal $s_{m_\ell}(t)$.

Figure 3.3–1 illustrates three different baseband signals and the corresponding data sequence. The first signal, called NRZ (non-return-to-zero), is the simplest. The binary information digit 1 is represented by a rectangular pulse of polarity A, and the binary digit 0 is represented by a rectangular pulse of polarity $-A$. Hence, the NRZ modulation is memoryless and is equivalent to a binary PAM or a binary PSK signal in a carrier-modulated system. The NRZI (non-return-to-zero, inverted) signal is different from the NRZ signal in that transitions from one amplitude level to another occur only when a 1 is transmitted. The amplitude level remains unchanged when a 0 is transmitted. This type of signal encoding is called *differential encoding*. The encoding operation is described mathematically by the relation

$$b_k = a_k \oplus b_{k-1} \tag{3.3–2}$$

where $\{a_k\}$ is the binary information sequence into the encoder, $\{b_k\}$ is the output sequence of the encoder, and $\oplus$ denotes addition modulo 2. When $b_k = 1$, the transmitted waveform is a rectangular pulse of amplitude A; and when $b_k = 0$, the transmitted waveform is a rectangular pulse of amplitude $-A$. Hence, the output of the encoder is mapped into one of two waveforms in exactly the same manner as for the NRZ signal. In other words, NRZI signaling can be considered as a differential encoder followed by an NRZ signaling scheme.

The existence of the differential encoder causes memory in NRZI signaling. Comparison of Equations 3.3–2 and 3.3–1 indicates that b_k can be considered as the state of the Markov chain. Since the information sequence is assumed to be binary, there are two states in the Markov chain, and the state transition diagram of the Markov chain is shown in Figure 3.3–2. The transition probabilities between states are determined by the probability of 0 and 1 generated by the source. If the source is equiprobable, all transition probabilities will be equal to $\frac{1}{2}$ and

$$\boldsymbol{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{3.3–3}$$

Using this $\boldsymbol{P}$, we can obtain the steady-state probability distribution as

$$\boldsymbol{p} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix} \tag{3.3–4}$$

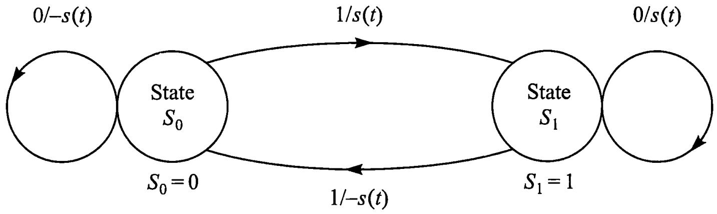


FIGURE 3.3–2
State transition diagram for NRZI signaling.

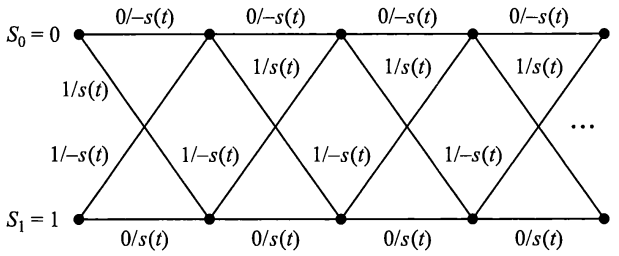


FIGURE 3.3–3
The trellis diagram for NRZI signaling.

We will use the steady-state probabilities to determine the power spectral density of modulation schemes with memory later in this chapter.

In general, if $\mathrm{P}[a_k = 1] = 1 - \mathrm{P}[a_k = 0] = p$, we have

$$\boldsymbol{P} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix} \tag{3.3–5}$$

The steady-state probability distribution in this case is again given by Equation 3.3–4.

Another way to display the memory introduced by the precoding operation is by means of a *trellis diagram*. The trellis diagram for the NRZI signal is illustrated in Figure 3.3–3. The trellis provides exactly the same information concerning the signal dependence as the state diagram, but also depicts a time evolution of the state transitions.

3.3–1 Continuous-Phase Frequency-Shift Keying (CPFSK)

In this section, we consider a class of digital modulation methods in which the phase of the signal is constrained to be continuous. This constraint results in a phase or frequency modulator that has memory.

As seen from Equation 3.2–54, a conventional FSK signal is generated by shifting the carrier by an amount $m\,\Delta f$, $1 \le m \le M$, to reflect the digital information that is being transmitted. This type of FSK signal was described in Section 3.2–4, and it is memoryless. The switching from one frequency to another may be accomplished by having $M = 2^k$ separate oscillators tuned to the desired frequencies and selecting one of the M frequencies according to the particular k-bit symbol that is to be transmitted in a signal interval of duration $T = k/R$ seconds. However, such abrupt switching from one oscillator output to another in successive signaling intervals results in relatively large spectral side lobes outside of the main spectral band of the signal; consequently, this method requires a large frequency band for transmission of the signal. To avoid the use of signals having large spectral side lobes, the information-bearing signal frequency modulates a single carrier whose frequency is changed continuously. The resulting frequency-modulated signal is phase-continuous, and hence, it is called *continuous-phase FSK* (CPFSK). This type of FSK signal has memory because the phase of the carrier is constrained to be continuous. To represent a CPFSK signal, we begin with a PAM signal

$$d(t) = \sum_n I_n g(t - nT) \tag{3.3–6}$$

where $\{I_n\}$ denotes the sequence of amplitudes obtained by mapping k-bit blocks of binary digits from the information sequence $\{a_n\}$ into the amplitude levels $\pm 1, \pm 3, \ldots, \pm(M-1)$ and $g(t)$ is a rectangular pulse of amplitude $1/2T$ and duration T seconds. The signal $d(t)$ is used to frequency-modulate the carrier. Consequently, the equivalent lowpass waveform $v(t)$ is expressed as

$$v(t) = \sqrt{\frac{2\mathcal{E}}{T}}\, e^{j\left[4\pi T f_d \int_{-\infty}^{t} d(\tau)\,d\tau + \phi_0\right]} \tag{3.3–7}$$

where f_d is the *peak frequency deviation* and ϕ_0 is the initial phase of the carrier. The carrier-modulated signal corresponding to Equation 3.3–7 may be expressed as

$$s(t) = \sqrt{\frac{2\mathcal{E}}{T}} \cos\left[2\pi f_c t + \phi(t; \boldsymbol{I}) + \phi_0\right] \tag{3.3–8}$$

where $\phi(t; \boldsymbol{I})$ represents the time-varying phase of the carrier, which is defined as

$$\begin{aligned} \phi(t; \boldsymbol{I}) &= 4\pi T f_d \int_{-\infty}^{t} d(\tau)\,d\tau \\ &= 4\pi T f_d \int_{-\infty}^{t} \left[\sum_n I_n g(\tau - nT)\right] d\tau \end{aligned} \tag{3.3–9}$$

Note that, although $d(t)$ contains discontinuities, the integral of $d(t)$ is continuous. Hence, we have a continuous-phase signal. The phase of the carrier in the interval $nT \le t \le (n+1)T$ is determined by integrating Equation 3.3–9. Thus,

$$\begin{aligned} \phi(t; \boldsymbol{I}) &= 2\pi f_d T \sum_{k=-\infty}^{n-1} I_k + 2\pi f_d q(t - nT) I_n \\ &= \theta_n + 2\pi h I_n q(t - nT) \end{aligned} \tag{3.3–10}$$

where h, θ_n, and $q(t)$ are defined as

$$h = 2 f_d T \tag{3.3–11}$$

$$\theta_n = \pi h \sum_{k=-\infty}^{n-1} I_k \tag{3.3–12}$$

$$q(t) = \begin{cases} 0 & t < 0 \\ \frac{t}{2T} & 0 \le t \le T \\ \frac{1}{2} & t > T \end{cases} \tag{3.3–13}$$

We observe that θ_n represents the accumulation (memory) of all symbols up to time $(n-1)T$. The parameter h is called the *modulation index*.

3.3–2 Continuous-Phase Modulation (CPM)

When expressed in the form of Equation 3.3–10, CPFSK becomes a special case of a general class of continuous-phase modulated (CPM) signals in which the carrier phase is

$$\phi(t; \boldsymbol{I}) = 2\pi \sum_{k=-\infty}^{n} I_k h_k q(t - kT), \qquad nT \le t \le (n+1)T \tag{3.3–14}$$

where $\{I_k\}$ is the sequence of M-ary information symbols selected from the alphabet $\pm 1, \pm 3, \ldots, \pm(M-1)$, $\{h_k\}$ is a sequence of modulation indices, and $q(t)$ is some normalized waveform shape. When $h_k = h$ for all k, the modulation index is fixed for all symbols. When the modulation index varies from one symbol to another, the signal is called multi-h CPM. In such a case, the $\{h_k\}$ are made to vary in a cyclic manner through a set of indices. The waveform $q(t)$ may be represented in general as the integral of some pulse $g(t)$, i.e.,

$$q(t) = \int_0^t g(\tau)\, d\tau \tag{3.3–15}$$

If $g(t) = 0$ for $t > T$, the signal is called *full-response CPM*. If $g(t) \neq 0$ for $t > T$, the modulated signal is called *partial-response CPM*. Figure 3.3–4 illustrates several pulse shapes for $g(t)$ and the corresponding $q(t)$. It is apparent that an infinite variety of CPM signals can be generated by choosing different pulse shapes $g(t)$ and by varying the modulation index h and the alphabet size M. We note that the CPM signal has memory that is introduced through the phase continuity.

Three popular pulse shapes are given in Table 3.3–1. LREC denotes a rectangular pulse of duration LT, where L is a positive integer. In this case, $L = 1$ results in a CPFSK signal, with the pulse as shown in Figure 3.3–4(a). The LREC pulse for $L = 2$ is shown in Figure 3.3–4(c). LRC denotes a raised cosine pulse of duration LT. The LRC pulses corresponding to $L = 1$ and $L = 2$ are shown in Figure 3.3–4(b) and (d), respectively. For $L > 1$, additional memory is introduced in the CPM signal by the pulse $g(t)$.

The third pulse given in Table 3.3–1 is called a *Gaussian minimum-shift keying* (GMSK) pulse with bandwidth parameter B, which represents the −3-dB bandwidth of the Gaussian pulse. Figure 3.3–4(e) illustrates a set of GMSK pulses with time-bandwidth products BT ranging from 0.1 to 1. We observe that the pulse duration increases as the bandwidth of the pulse decreases, as expected. In practical applications, the pulse is usually truncated to some specified fixed duration. GMSK with $BT = 0.3$ is used in the European digital cellular communication system, called GSM. From Figure 3.3–4(e) we observe that when $BT = 0.3$, the GMSK pulse may be truncated at $|t| = 1.5T$ with a relatively small error incurred for $t > 1.5T$.

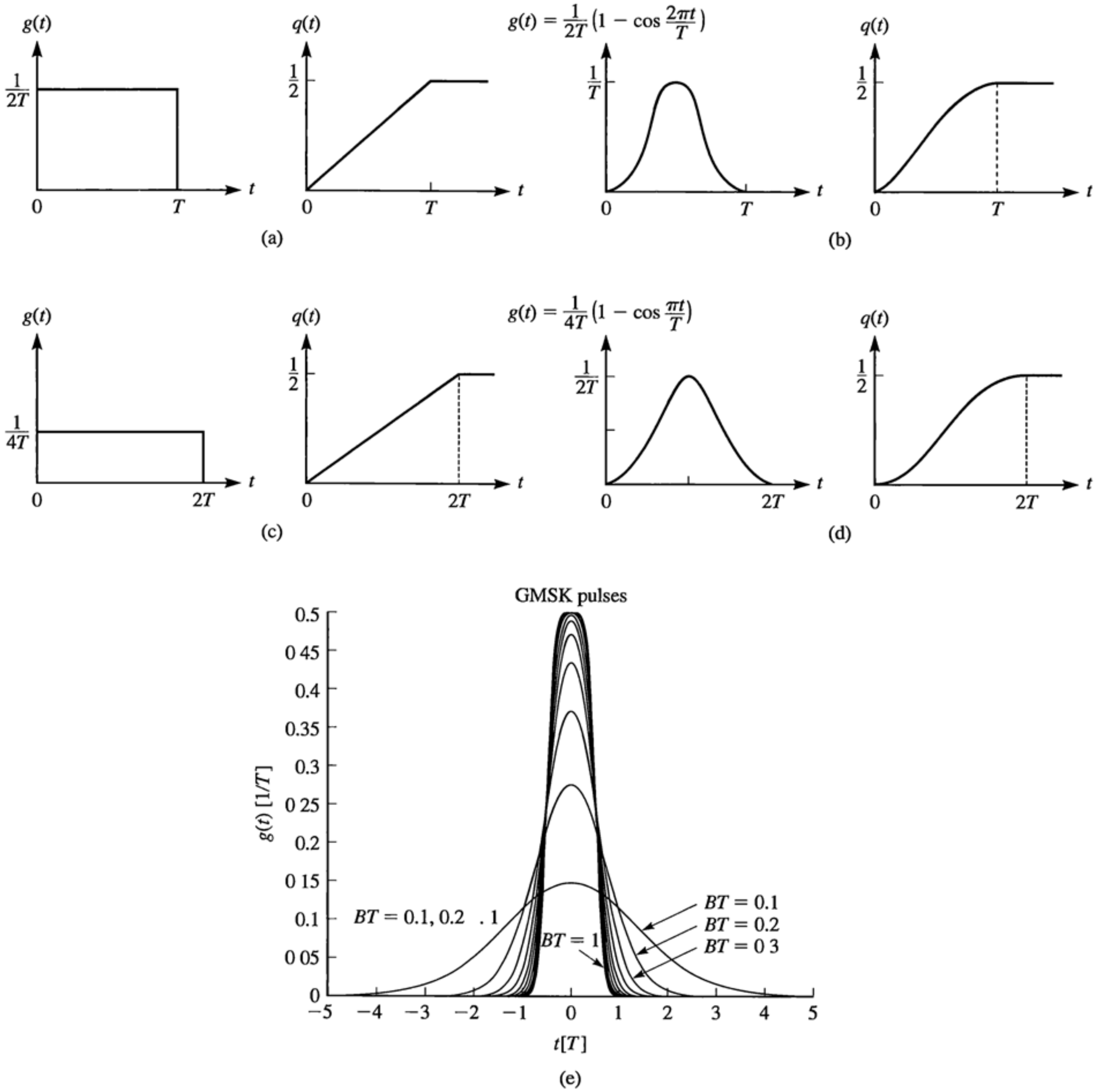


FIGURE 3.3–4
Pulse shapes for full-response CPM (a, b) and partial-response CPM (c, d), and GMSK (e).

TABLE 3.3–1
Some Commonly Used CPM Pulse Shapes

LREC	$g(t) = \begin{cases} \frac{1}{2LT} & 0 \le t \le LT \\ 0 & \text{otherwise} \end{cases}$
LRC	$g(t) = \begin{cases} \frac{1}{2LT}\left(1 - \cos\frac{2\pi t}{LT}\right) & 0 \le t \le LT \\ 0 & \text{otherwise} \end{cases}$
GMSK	$g(t) = \frac{Q\left(2\pi B\left(t - \frac{T}{2}\right)\right) - Q\left(2\pi B\left(t + \frac{T}{2}\right)\right)}{\sqrt{\ln 2}}$

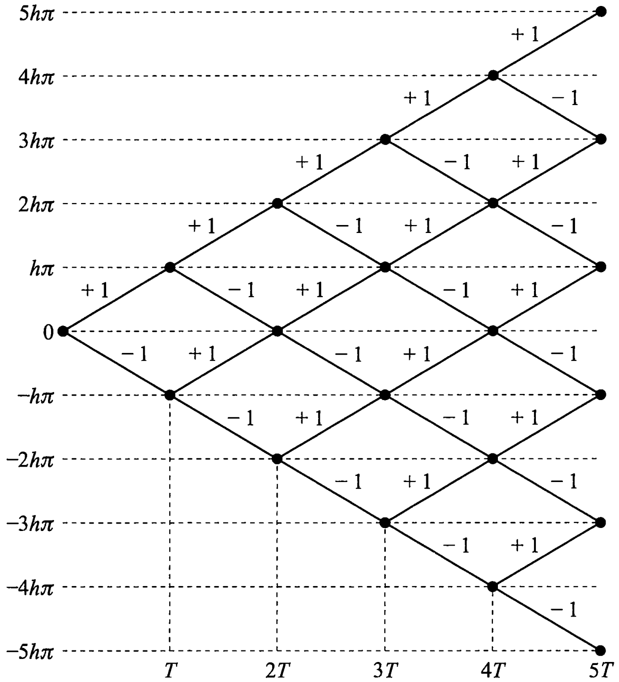


FIGURE 3.3–5
Phase trajectory for binary CPFSK.

It is instructive to sketch the set of phase trajectories $\phi(t; \boldsymbol{I})$ generated by all possible values of the information sequence $\{I_n\}$. For example, in the case of CPFSK with binary symbols $I_n = \pm 1$, the set of phase trajectories beginning at time $t = 0$ is shown in Figure 3.3–5. For comparison, the phase trajectories for quaternary CPFSK are illustrated in Figure 3.3–6.

These phase diagrams are called *phase trees*. We observe that the phase trees for CPFSK are piecewise linear as a consequence of the fact that the pulse $g(t)$ is rectangular. Smoother phase trajectories and phase trees are obtained by using pulses that do not contain discontinuities, such as the class of raised cosine pulses. For example, a phase trajectory generated by the sequence $(1, -1, -1, -1, 1, 1, -1, 1)$ for a partial-response, raised cosine pulse of length $3T$ is illustrated in Figure 3.3–7. For comparison, the corresponding phase trajectory generated by CPFSK is also shown.

The phase trees shown in these figures grow with time. However, the phase of the carrier is unique only in the range from $\phi = 0$ to $\phi = 2\pi$ or, equivalently, from $\phi = -\pi$ to $\phi = \pi$. When the phase trajectories are plotted modulo 2π, say, in the range $(-\pi, \pi)$, the phase tree collapses into a structure called a *phase trellis*. To properly view the phase trellis diagram, we may plot the two quadrature components $x_i(t; \boldsymbol{I}) = \cos\phi(t; \boldsymbol{I})$ and $x_q(t; \boldsymbol{I}) = \sin\phi(t; \boldsymbol{I})$ as functions of time. Thus, we generate a three-dimensional plot in which the quadrature components x_i and x_q appear on the surface of a cylinder of unit radius. For example, Figure 3.3–8 illustrates the phase trellis or phase cylinder obtained with binary modulation, a modulation index $h = \frac{1}{2}$, and a raised cosine pulse of length $3T$.

Simpler representations for the phase trajectories can be obtained by displaying only the terminal values of the signal phase at the time instants $t = nT$. In this case, we restrict the modulation index of the CPM signal to be rational. In particular, let us assume that $h = m/p$, where m and p are relatively prime integers. Then a full-response

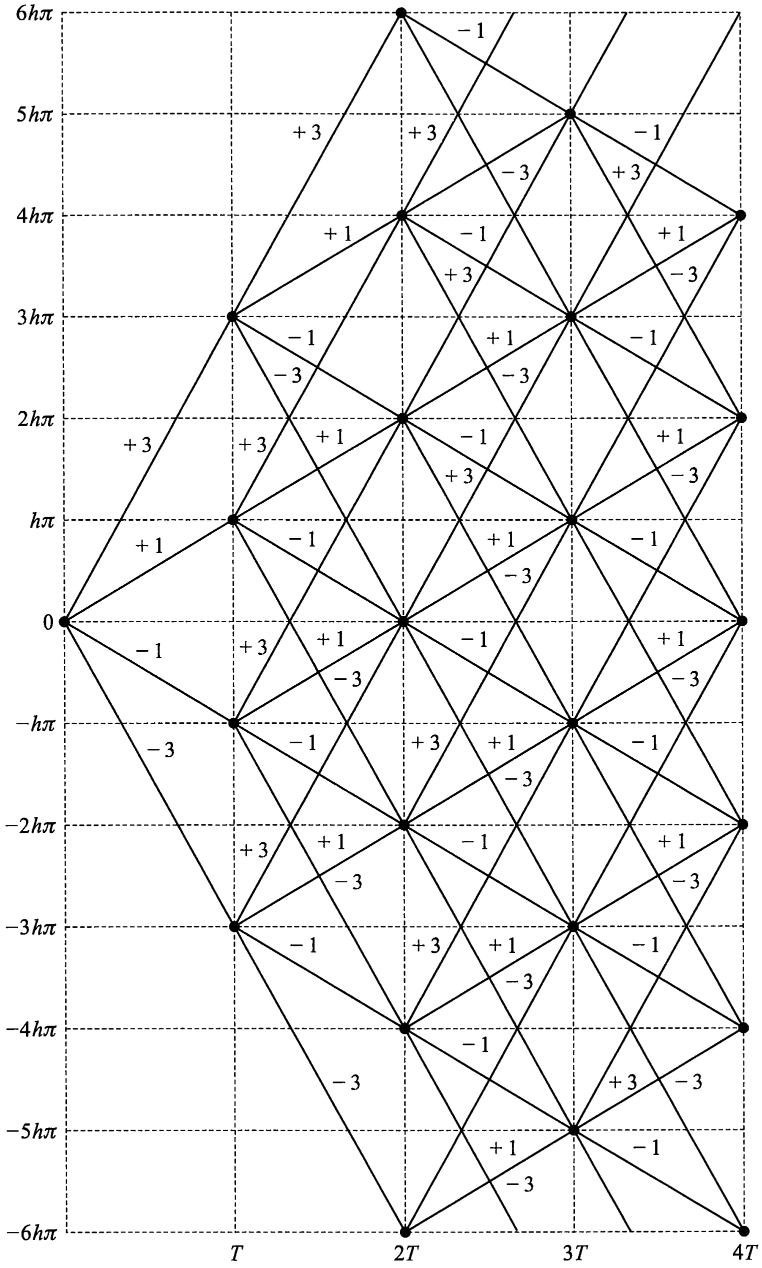

6hπ
5hπ
4hπ
3hπ
2hπ
hπ
0
−hπ
−2hπ
−3hπ
−4hπ
−5hπ
−6hπ
T
2T
3T
4T
+3
+1
−1
−3

FIGURE 3.3–6
Phase trajectory for quaternary CPFSK.

CPM signal at the time instants $t = nT$ will have the terminal phase states

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \cdots, \frac{(p-1)\pi m}{p}\right\} \tag{3.3–16}$$

when m is even and

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \cdots, \frac{(2p-1)\pi m}{p}\right\} \tag{3.3–17}$$

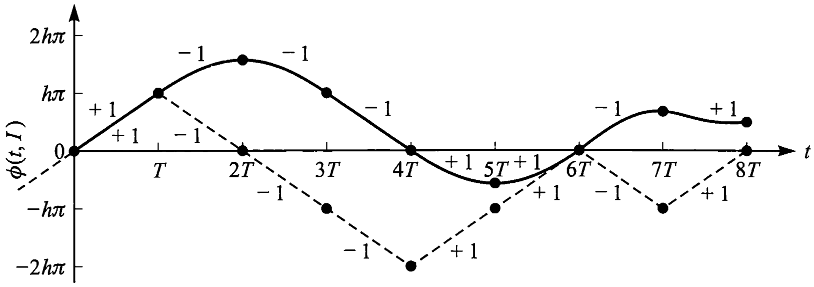


FIGURE 3.3–7

Phase trajectories for binary CPFSK (dashed) and binary, partial-response CPM based on raised cosine pulse of length $3T$ (solid). [*Source:* Sundberg (1986), © 1986 IEEE]

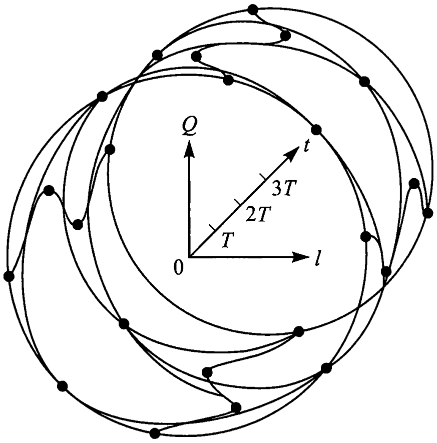


FIGURE 3.3–8

Phase cylinder for binary CPM with $h = \frac{1}{2}$ and a raised cosine pulse of length $3T$. [*Source:* Sundberg (1986), © 1986 IEEE]

when m is odd. Hence, there are p terminal phase states when m is even and $2p$ states when m is odd. On the other hand, when the pulse shape extends over L symbol intervals (partial-response CPM), the number of phase states may increase up to a maximum of S_t, where

$$S_t = \begin{cases} pM^{L-1} & \text{even } m \\ 2pM^{L-1} & \text{odd } m \end{cases} \tag{3.3–18}$$

where M is the alphabet size. For example, the binary CPFSK signal (full-response, rectangular pulse) with $h = \frac{1}{2}$ has $S_t = 4$ (terminal) phase states. The *state trellis* for this signal is illustrated in Figure 3.3–9. We emphasize that the phase transitions from one state to another are not true phase trajectories. They represent phase transitions for the (terminal) states at the time instants $t = nT$.

An alternative representation to the state trellis is the state diagram, which also illustrates the state transitions at the time instants $t = nT$. This is an even more compact representation of the CPM signal characteristics. Only the possible (terminal) phase states and their transitions are displayed in the state diagram. Time does not appear explicitly as a variable. For example, the state diagram for the CPFSK signal with $h = \frac{1}{2}$ is shown in Figure 3.3–10.

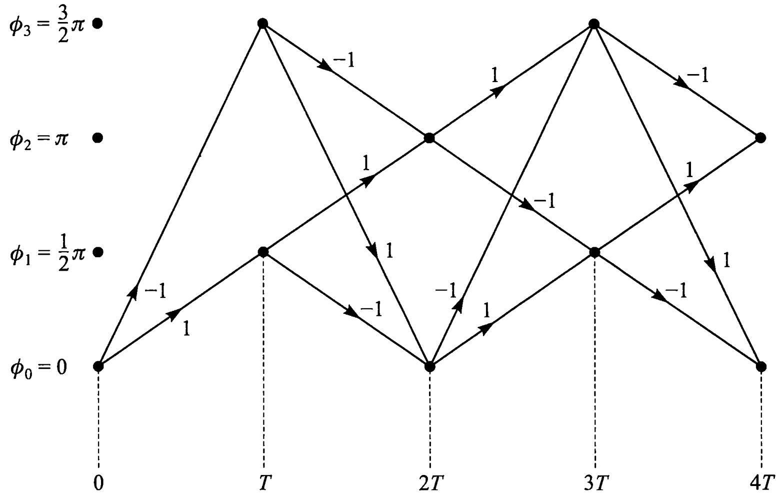


FIGURE 3.3–9
State trellis for binary CPFSK with $h = \frac{1}{2}$.

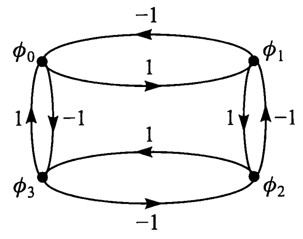


FIGURE 3.3–10
State diagram for binary CPFSK with $h = \frac{1}{2}$.

Minimum-Shift Keying (MSK)

MSK is a special form of binary CPFSK (and, therefore, CPM) in which the modulation index $h = \frac{1}{2}$ and $g(t)$ is a rectangular pulse of duration T. The phase of the carrier in the interval $nT \le t \le (n+1)T$ is

$$\begin{aligned}\phi(t; \boldsymbol{I}) &= \frac{1}{2}\pi \sum_{k=-\infty}^{n-1} I_k + \pi I_n q(t - nT) \\ &= \theta_n + \frac{1}{2}\pi I_n \left(\frac{t - nT}{T}\right), \qquad nT \le t \le (n+1)T\end{aligned} \tag{3.3–19}$$

and the modulated carrier signal is

$$\begin{aligned}s(t) &= A \cos\left[2\pi f_c t + \theta_n + \frac{1}{2}\pi I_n \left(\frac{t - nT}{T}\right)\right] \\ &= A \cos\left[2\pi \left(f_c + \frac{1}{4T} I_n\right) t - \frac{1}{2} n\pi I_n + \theta_n\right], \qquad nT \le t \le (n+1)T\end{aligned} \tag{3.3–20}$$

Equation 3.3–20 indicates that the binary CPFSK signal can be expressed as a sinusoid having one of two possible frequencies in the interval $nT \le t \le (n+1)T$. If we define these frequencies as

$$f_1 = f_c - \frac{1}{4T}$$
$$f_2 = f_c + \frac{1}{4T} \tag{3.3–21}$$

then the binary CPFSK signal given by Equation 3.3–20 may be written in the form

$$s_i(t) = A\cos\left[2\pi f_i t + \theta_n + \frac{1}{2}n\pi(-1)^{i-1}\right], \qquad i = 1, 2 \tag{3.3–22}$$

which represents an FSK signal with frequency separation of $\Delta f = f_2 - f_1 = 1/2T$. From the discussion following Equation 3.2–58 we recall that $\Delta f = 1/2T$ is the minimum frequency separation that is necessary to ensure the orthogonality of signals $s_1(t)$ and $s_2(t)$ over a signaling interval of length T. This explains why binary CPFSK with $h = \frac{1}{2}$ is called minimum-shift keying (MSK). The phase in the nth signaling interval is the phase state of the signal that results in phase continuity between adjacent intervals.

Offset QPSK (OQPSK)

Consider the QPSK system with constellation shown in Figure 3.3–11. In this system each 2 information bits is mapped into one of the constellation points. The constellation and one possible mapping of bit sequences of length 2 are shown in Figure 3.3–11.

Now assume we are interested in transmitting the binary sequence 11000111. To do this, we can split this sequence into binary sequences 11, 00, 01, and 11 and transmit the corresponding points in the constellation. The first bit in each binary sequence determines the in-phase (I) component of the baseband signal with a duration $2T_b$, and the second bit determines the quadrature (Q) component of it, again of duration $2T_b$. The in-phase and quadrature components for this bit sequence are shown in Figure 3.3–12. Note that changes can occur only at even multiples of T_b, and there are instances at which both I and Q components change simultaneously, resulting in a change of 180° in the phase, for instance, at $t = 2T_b$ in Figure 3.3–12. The possible phase transitions for QPSK signals, that can occur only at time instances of the form nT_b, where n is even, are shown in Figure 3.3–13.

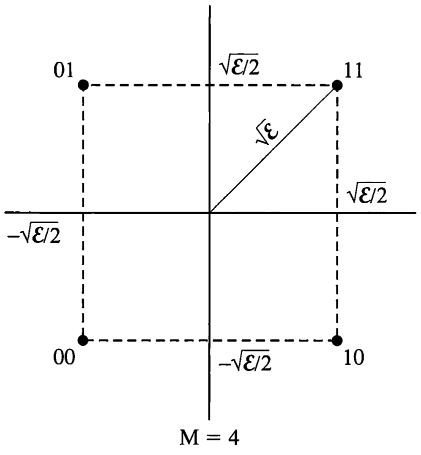


FIGURE 3.3–11
A possible mapping for QPSK signal.

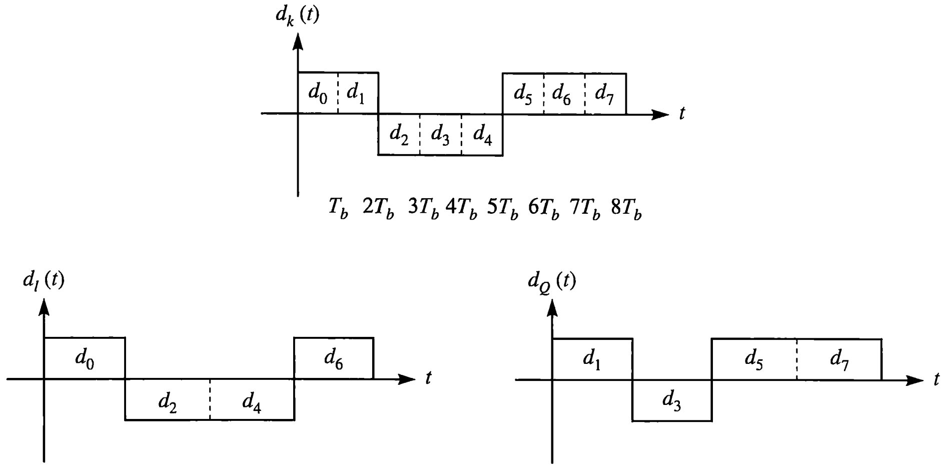


FIGURE 3.3–12
The in-phase and quadrature components for QPSK.

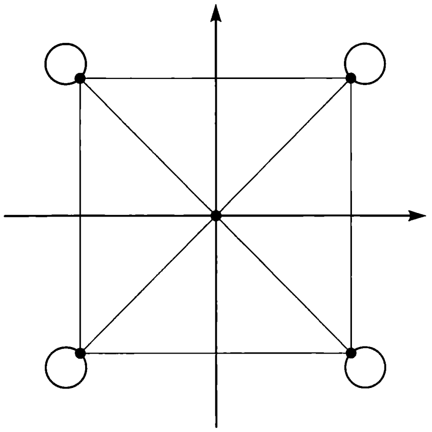


FIGURE 3.3–13
Possible phase transitions in QPSK signaling.

To prevent 180° phase changes that cause abrupt changes in the signal, resulting in large spectral side lobes, a version of QPSK, known as *offset QPSK* (OQPSK), or *staggered QPSK* (SQPSK), is introduced. In OQPSK, the in-phase and quadrature components of the standard QPSK are misaligned by T_b. The in-phase and quadrature components for the sequence 11000111 are shown in Figure 3.3–14. Misalignment of the in-phase and quadrature components prevents both components changing at the same time and thus prevents phase transitions of 180°. This reduces the abrupt jumps in the modulated signal. The absence of 180° phase jump is, however, offset by more frequent ±90° phase shifts. The overall effect is that, as we will see later, standard QPSK and OQPSK have the same power spectral density. The phase transition diagram for OQPSK is shown in Figure 3.3–15.

The OQPSK signal can be written as

$$
\begin{aligned}
s(t) = A \Bigg[& \left(\sum_{n=-\infty}^{\infty} I_{2n} g(t - 2nT) \right) \cos 2\pi f_c t \\
& + \left(\sum_{n=-\infty}^{\infty} I_{2n+1} g(t - 2nT - T) \right) \sin 2\pi f_c t \Bigg]
\end{aligned}
\tag{3.3–23}
$$

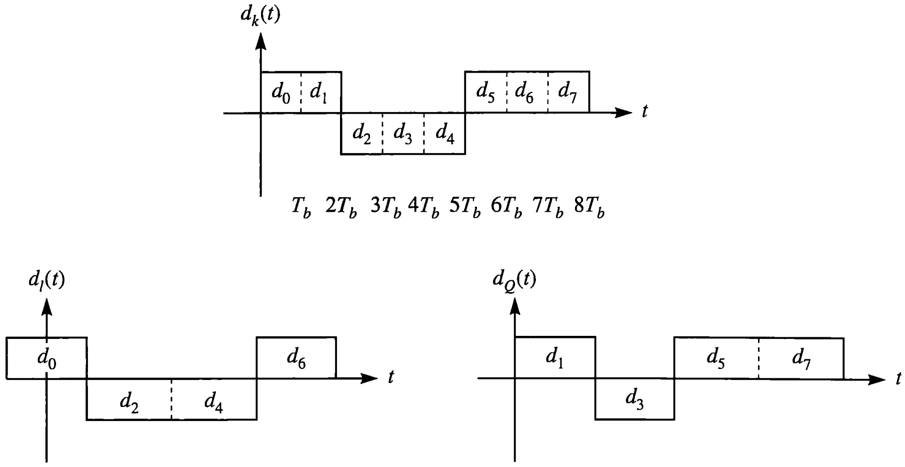


FIGURE 3.3–14
The in-phase and quadrature components for OQPSK signaling.

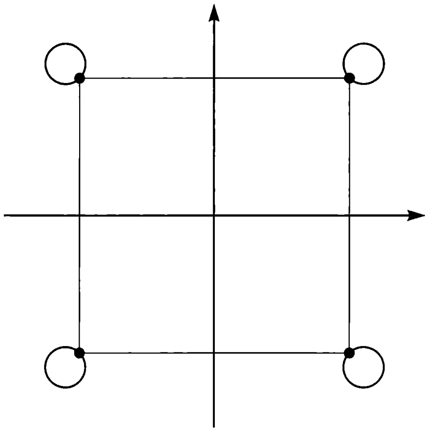


FIGURE 3.3–15
Phase transition diagram for OQPSK signaling.

with the lowpass equivalent of

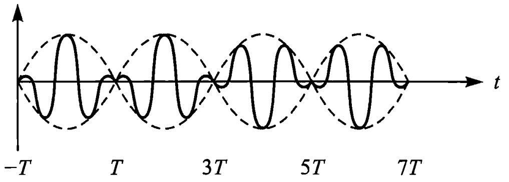
$$s_l(t) = A\left[\sum_{n=-\infty}^{\infty} I_{2n}g(t-2nT)\right] - j\left[\sum_{n=-\infty}^{\infty} I_{2n+1}g(t-2nT-T)\right] \tag{3.3–24}$$

MSK may also be represented as a form of OQPSK. Specifically, we may express (see Problem 3.26 and Example 3.3–1) the equivalent lowpass digitally modulated MSK signal in the form of Equation 3.3–24 with

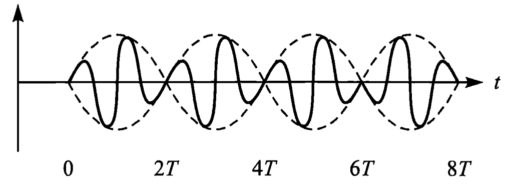
$$g(t) = \begin{cases} \sin\frac{\pi t}{2T} & 0 \le t \le 2T \\ 0 & \text{otherwise} \end{cases} \tag{3.3–25}$$

Figure 3.3–16 illustrates the representation of an MSK signal as two staggered quadrature-modulated binary PSK signals. The corresponding sum of the two quadrature signals is a constant-amplitude, frequency-modulated signal.

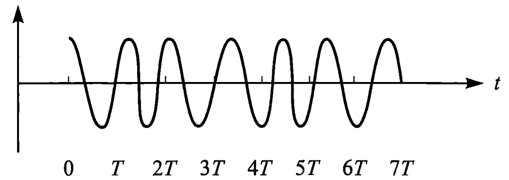
It is also interesting to compare the waveforms for MSK with offset QPSK in which the pulse $g(t)$ is rectangular for $0 \le t \le 2T$, and with conventional QPSK in which the pulse $g(t)$ is rectangular for $0 \le t \le 2T$. Clearly, all three of the modulation methods



(a) In-phase signal component



(b) Quadrature signal component



(c) MSK signal [sum of (*a*) and (*b*)]

FIGURE 3.3–16
Representation of MSK as an OQPSK signal with a sinusoidal envelope.

result in identical data rates. The MSK signal has continuous phase; therefore, there exist no jumps in its waveform. However, since it is essentially a frequency modulation system, there are jumps in its instantaneous frequency. The offset QPSK signal with a rectangular pulse is basically two binary PSK signals for which the phase transitions are staggered in time by T seconds. Thus, the signal contains phase jumps of $\pm 90^\circ$ that may occur as often as every T seconds. OQPSK is a signaling scheme with constant frequency, but there exist jumps in its waveform. On the other hand, the conventional four-phase PSK signal with constant amplitude will contain phase jumps of $\pm 180^\circ$ or $\pm 90^\circ$ every $2T$ seconds. An illustration of these three signal types is given in Figure 3.3–17.

QPSK signaling with rectangular pulses has constant envelope, but in practice filtered pulse shapes like the raised cosine signal are preferred and are more widely employed. When filtered pulse shapes are used, the QPSK signal will not be a constant-envelope modulation scheme, and the 180° phase shifts cause the envelope to pass through zero. Nonconstant envelope signals are not desirable particularly when used with nonlinear devices such as class C amplifiers or TWTs. In such cases OQPSK is a useful alternative to QPSK.

In MSK the phase is continuous—since it is a special case of CPFSK—but the frequency has jumps in it. If these jumps are smoothed, the spectrum will be more compact. GMSK signaling discussed earlier in this chapter and summarized in Table 3.3–1 is a signaling scheme that addresses this problem by shaping the lowpass binary signal before being applied to the MSK modulator and therefore results in smoother transitions in frequency between signaling intervals. This results in more compact spectral characteristics. The baseband signal is shaped in GMSK, but since the shaping occurs before modulation, the resulting modulated signal will be of constant envelope.

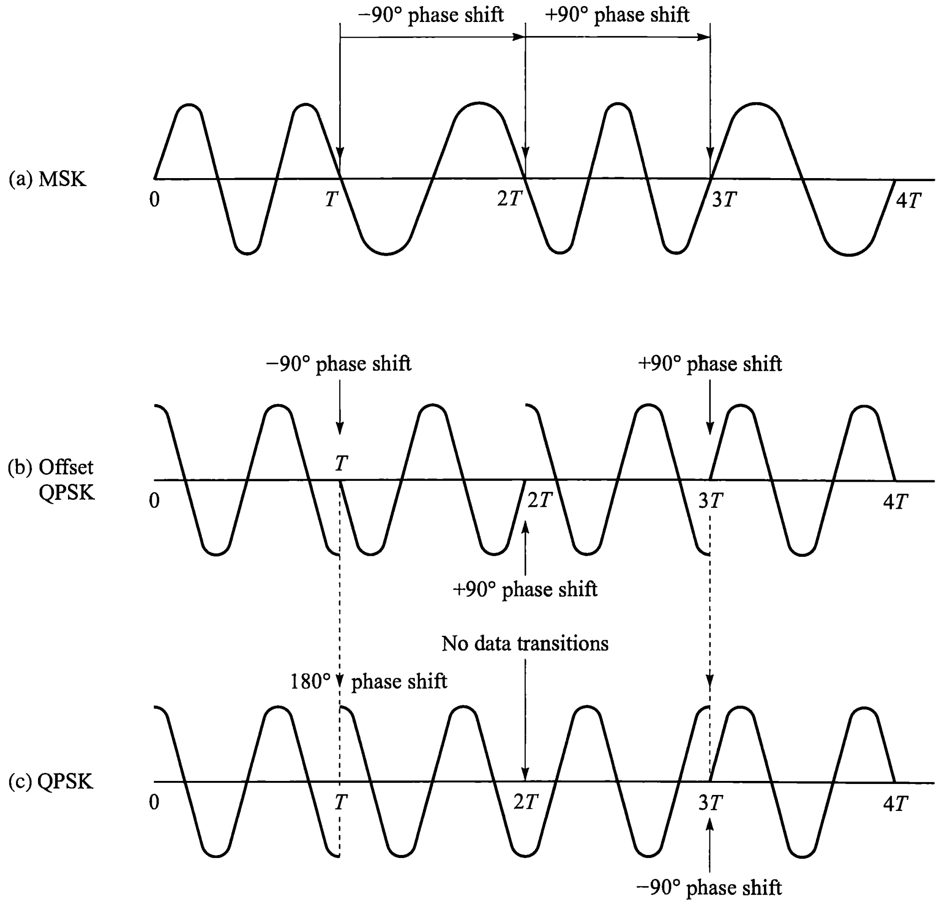

−90° phase shift
+90° phase shift
(a) MSK
0
T
2T
3T
4T
−90° phase shift
+90° phase shift
(b) Offset QPSK
T
0
2T
3T
4T
+90° phase shift
No data transitions
180° phase shift
(c) QPSK
0
T
2T
3T
4T
−90° phase shift

FIGURE 3.3–17
MSK, OQPSK, and QPSK signals.

Linear Representation of CPM Signals

As described above, CPM is a nonlinear modulation technique with memory. However, CPM may also be represented as a linear superposition of signal waveforms. Such a representation provides an alternative method for generating the modulated signal at the transmitter and/or demodulating the signal at the receiver. Following the development originally given by Laurent (1986), we demonstrate that binary CPM may be represented by a linear superposition of a finite number of amplitude-modulated pulses, provided that the pulse $g(t)$ is of finite duration LT, where T is the bit interval. We begin with the equivalent lowpass representation of CPM, which is

$$v(t) = \sqrt{\frac{2\mathcal{E}}{T}} e^{j\phi(t;\boldsymbol{I})}, \qquad nT \le t \le (n+1)T \tag{3.3–26}$$

where

$$\begin{aligned} \phi(t;\boldsymbol{I}) &= 2\pi h \sum_{k=-\infty}^{n} I_k q(t-kT), \qquad nT \le t \le (n+1)T \\ &= \pi h \sum_{k=-\infty}^{n-L} I_k + 2\pi h \sum_{k=n-L+1}^{n} I_k q(t-kT) \end{aligned} \tag{3.3–27}$$

and $q(t)$ is the integral of the pulse $g(t)$, as previously defined in Equation 3.3–15. The exponential term $\exp[j\phi(t;\,\boldsymbol{I})]$ may be expressed as

$$\exp[j\phi(t;\,\boldsymbol{I})] = \exp\left(j\pi h \sum_{k=-\infty}^{n-L} I_k\right) \prod_{k=0}^{L-1} \exp\{j2\pi h I_{n-k} q\,[t-(n-k)T]\} \quad (3.3\text{–}28)$$

Note that the first term on the right-hand side of Equation 3.3–28 represents the cumulative phase up to the information symbol I_{n-L}, and the second term consists of a product of L phase terms. Assuming that the modulation index h is not an integer and the data symbols are binary, i.e., $I_k = \pm 1$, the kth phase term may be expressed as

$$\begin{aligned}\exp\{j2\pi h I_{n-k} q\,[t-(n-k)T]\} &= \frac{\sin \pi h}{\sin \pi h} \exp\{j2\pi h I_{n-k} q\,[t-(n-k)T]\} \\ &= \frac{\sin\{\pi h - 2\pi h q[t-(n-k)]T\}}{\sin \pi h} \\ &\quad + \exp(j\pi h I_{n-k}) \frac{\sin\{2\pi h q[t-(n-k)T]\}}{\sin \pi h}\end{aligned} \quad (3.3\text{–}29)$$

It is convenient to define the signal pulse $s_0(t)$ as

$$s_0(t) = \begin{cases} \frac{\sin 2\pi h q(t)}{\sin \pi h} & 0 \le t \le LT \\ \frac{\sin[\pi h - 2\pi h q(t-LT)]}{\sin \pi h} & LT \le t \le 2LT \\ 0 & \text{otherwise} \end{cases} \quad (3.3\text{–}30)$$

Then

$$\begin{aligned}\exp[j\phi(t;\,\boldsymbol{I})] = \exp\left(j\pi h \sum_{k=-\infty}^{n-L} I_k\right) \prod_{k=0}^{L-1} \{s_0[t+(k+L-n)T] \\ + \exp(j\pi h I_{n-k}) s_0[t-(k-n)T]\}\end{aligned} \quad (3.3\text{–}31)$$

By performing the multiplication over the L terms in the product, we obtain a sum of 2^L terms, where 2^{L-1} terms are distinct and the other 2^{L-1} terms are time-shifted versions of the distinct terms. The final result may be expressed as

$$\exp[j\phi(t;\,\boldsymbol{I})] = \sum_n \sum_{k=0}^{2^{L-1}-1} e^{j\pi h A_{k,n}} c_k(t-nT) \quad (3.3\text{–}32)$$

where the pulses $c_k(t)$, for $0 \le k \le 2^{L-1}-1$, are defined as

$$c_k(t) = s_0(t) \prod_{n=1}^{L-1} s_0[t+(n+La_{k,n})T], \qquad 0 \le t \le T \times \min_n [L(2-a_{k,n})-n] \quad (3.3\text{–}33)$$

and each pulse is weighted by a complex coefficient $\exp(j\pi h A_{k,n})$, where

$$A_{k,n} = \sum_{m=-\infty}^{n} I_m - \sum_{m=1}^{L-1} I_{n-m} a_{k,m} \quad (3.3\text{–}34)$$

and the $\{a_{k,n} = 0 \text{ or } 1\}$ are the coefficients in the binary representation of the index k, i.e.,

$$k = \sum_{m=1}^{L-1} 2^{m-1} a_{k,m}, \qquad k = 0, 1, \ldots, 2^{L-1} - 1 \tag{3.3–35}$$

Thus, the binary CPM signal is expressed as a weighted sum of 2^{L-1} real-valued pulses $\{c_k(t)\}$.

In this representation of CPM as a superposition of amplitude-modulated pulses, the pulse $c_0(t)$ is the most important component, because its duration is the longest and it contains the most significant part of the signal energy. Consequently, a simple approximation to a CPM signal is a partial-response PAM signal having $c_0(t)$ as the basic pulse shape.

The focus for the above development was binary CPM. A representation of M-ary CPM as a superposition of PAM waveforms has been described by Mengali and Morelli (1995).

EXAMPLE 3.3–1. As a special case, let us consider the MSK signal, for which $h = \frac{1}{2}$ and $g(t)$ is a rectangular pulse of duration T. In this case,

$$\begin{aligned}\phi(t; \boldsymbol{I}) &= \frac{\pi}{2} \sum_{k=-\infty}^{n-1} I_k + \pi I_n q(t - nT) \\ &= \theta_n + \frac{\pi}{2} I_n \left(\frac{t - nT}{T} \right), \qquad nT \le t \le (n+1)T\end{aligned}$$

and

$$\exp[j\phi(t; \boldsymbol{I})] = \sum_n b_n c_0(t - nT)$$

where

$$c_0(t) = \begin{cases} \sin \frac{\pi t}{2T} & 0 \le t \le 2T \\ 0 & \text{otherwise} \end{cases}$$

and

$$b_n = e^{j\pi A_{0,n}/2} = e^{j\pi(\theta_n + I_n)/2}$$

The complex-valued modified data sequence $\{b_n\}$ may be expressed recursively as

$$b_n = j b_{n-1} I_n$$

so that b_n alternates in taking real and imaginary values. By separating the real and the imaginary components, we obtain the equivalent lowpass signal representation given by Equations 3.3–24 and 3.3–25.

3.4

POWER SPECTRUM OF DIGITALLY MODULATED SIGNALS

In this section we study the power spectral density of digitally modulated signals. The information about the power spectral density helps us determine the required transmission bandwidth of these modulation schemes and their bandwidth efficiency. We start by considering a general modulation scheme with memory in which the current transmitted signal can depend on the entire history of the information sequence and then specialize this general formulation to the cases where the modulation system has a finite memory, the case where the modulation is linear, and when the modulated signal can be determined by the state of a Markov chain. We conclude this section with the spectral characteristics of CPM and CPFSK signals.

3.4–1 Power Spectral Density of a Digitally Modulated Signal with Memory

Here we assume that the bandpass modulated signal is denoted by $v(t)$ with a lowpass equivalent signal of the form

$$v_l(t) = \sum_{n=-\infty}^{\infty} s_l(t - nT; \boldsymbol{I}_n) \tag{3.4–1}$$

Here $s_l(t; \boldsymbol{I}_n) \in \{s_{1l}(t), s_{2l}(t), \ldots, s_{Ml}(t)\}$ is one of the possible M lowpass equivalent signals determined by the information sequence up to time n, denoted by $\boldsymbol{I}_n = (\ldots, I_{n-2}, I_{n-1}, I_n)$. We assume that I_n is stationary process. Our goal here is to determine the power spectral density of $v(t)$. This is done by first deriving the power spectral density of $v_l(t)$ and using Equation 2.9–14 to obtain the power spectral density of $v(t)$.

We first determine the autocorrelation function of $v_l(t)$.

$$\begin{aligned} R_{v_l}(t+\tau, t) &= \mathrm{E}\left[v_l(t+\tau)v_l^*(t)\right] \\ &= \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \mathrm{E}\left[s_l(t+\tau-nT; \boldsymbol{I}_n)s_l^*(t-mT; \boldsymbol{I}_m)\right] \end{aligned} \tag{3.4–2}$$

Changing t to $t+T$ does not change the mean and the autocorrelation function of $v_l(t)$, hence $v_l(t)$ is a cyclostationary process; to determine its power spectral density, we have to average $R_{v_l}(t+\tau, t)$ over one period T. We have (with a change of variable of $k = n - m$)

$$\begin{aligned} \bar{R}_{v_l}(\tau) &= \frac{1}{T} \sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \int_0^T \mathrm{E}\left[s_l(t+\tau-mT-kT; \boldsymbol{I}_{m+k})s_l^*(t-mT; \boldsymbol{I}_m)\right] dt \\ &\overset{(a)}{=} \frac{1}{T} \sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \int_{-mT}^{-(m-1)T} \mathrm{E}\left[s_l(u+\tau-kT; \boldsymbol{I}_k)s_l^*(u; \boldsymbol{I}_0)\right] du \\ &= \frac{1}{T} \sum_{k=-\infty}^{\infty} \int_{-\infty}^{\infty} \mathrm{E}\left[s_l(u+\tau-kT; \boldsymbol{I}_k)s_l^*(u; \boldsymbol{I}_0)\right] du \end{aligned} \tag{3.4–3}$$

where in (a) we have introduced a change of variable of the form $u = t - mT$ and we have used the fact that the Markov chain is in the steady state and the input process $\{I_n\}$ is stationary. Defining

$$g_k(\tau) = \int_{-\infty}^{\infty} \mathrm{E}\left[s_l(t+\tau; \boldsymbol{I}_k)s_l^*(t; \boldsymbol{I}_0)\right] dt \tag{3.4–4}$$

we can write Equation 3.4–3 as

$$\bar{R}_{v_l}(\tau) = \frac{1}{T}\sum_{k=-\infty}^{\infty} g_k(\tau - kT) \tag{3.4–5}$$

The power spectral density of $v_l(t)$, which is the Fourier transform of $R_{v_l}(\tau)$, is therefore given by

$$\begin{aligned} \mathcal{S}_{v_l}(f) &= \frac{1}{T}\,\mathscr{F}\left[\sum_k g_k(\tau - kT)\right] \\ &= \frac{1}{T}\sum_{k=-\infty}^{\infty} G_k(f)e^{-j2\pi kfT} \end{aligned} \tag{3.4–6}$$

where $G_k(f)$ denotes the Fourier transform of $g_k(\tau)$. We can also express $G_k(f)$ in the following form:

$$\begin{aligned} G_k(f) &= \mathscr{F}\left[\int_{-\infty}^{\infty} \mathrm{E}\left[s_l(t+\tau; \boldsymbol{I}_k)s_l^*(t; \boldsymbol{I}_0)\right] dt\right] \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{E}\left[s_l(t+\tau; \boldsymbol{I}_k)s_l^*(t; \boldsymbol{I}_0)\right] e^{-j2\pi f\tau}\,dt\,d\tau \\ &= \mathrm{E}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s_l(t+\tau; \boldsymbol{I}_k)e^{-j2\pi f(t+\tau)}\,s_l^*(t; \boldsymbol{I}_0)e^{j2\pi ft}\,dt\,d\tau\right] \\ &= \mathrm{E}\left[S_l(f; \boldsymbol{I}_k)S_l^*(f; \boldsymbol{I}_0)\right] \end{aligned} \tag{3.4–7}$$

where $S_l(f; \boldsymbol{I}_k)$ and $S_l(f; \boldsymbol{I}_0)$ are Fourier transforms of $s_l(t; \boldsymbol{I}_k)$ and $s_l(t; \boldsymbol{I}_0)$, respectively.

From Equation 3.4–7, we conclude that $G_0(f) = \mathrm{E}\left[|S_l(f; \boldsymbol{I}_0)|^2\right]$ is real, and $G_{-k}(f) = G_k^*(f)$ for $k \geq 1$. If we define

$$G'_k(f) = G_k(f) - G_0(f) \tag{3.4–8}$$

we can readily see that

$$\begin{aligned} G'_{-k}(f) &= G_k'^*(f) \\ G'_0(f) &= 0 \end{aligned} \tag{3.4–9}$$

Equation 3.4–6 can be written as

$$\begin{aligned}
\mathcal{S}_{v_l}(f) &= \frac{1}{T}\sum_{k=-\infty}^{\infty}\left(G_k(f)-G_0(f)\right)e^{-j2\pi kfT}+\frac{1}{T}\sum_{k=-\infty}^{\infty}G_0(f)e^{-j2\pi kfT}\\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}G_k'(f)e^{-j2\pi kfT}+\frac{1}{T^2}\sum_{k=-\infty}^{\infty}G_0(f)\,\delta\left(f-\frac{k}{T}\right)\\
&= \frac{2}{T}\operatorname{Re}\left[\sum_{k=1}^{\infty}G_k'(f)e^{-j2\pi kfT}\right]+\frac{1}{T^2}\sum_{k=-\infty}^{\infty}G_0\left(\frac{k}{T}\right)\delta\left(f-\frac{k}{T}\right)\\
&= \mathcal{S}_{v_l}^{(c)}(f)+\mathcal{S}_{v_l}^{(d)}(f)
\end{aligned} \qquad (3.4\text{–}10)$$

where we have used Equation 3.4–9 and the well-known relation

$$\sum_{k=-\infty}^{\infty}e^{j2\pi kfT}=\frac{1}{T}\sum_{k=-\infty}^{\infty}\delta\left(f-\frac{k}{T}\right) \qquad (3.4\text{–}11)$$

$\mathcal{S}_{v_l}^{(c)}(f)$ and $\mathcal{S}_{v_l}^{(d)}(f)$, defined by

$$\begin{aligned}
\mathcal{S}_{v_l}^{(c)}(f) &= \frac{2}{T}\operatorname{Re}\left[\sum_{k=1}^{\infty}G_k'(f)e^{-j2\pi kfT}\right]\\
\mathcal{S}_{v_l}^{(d)}(f) &= \frac{1}{T^2}\sum_{k=-\infty}^{\infty}G_0\left(\frac{k}{T}\right)\delta\left(f-\frac{k}{T}\right)
\end{aligned} \qquad (3.4\text{–}12)$$

represent the *continuous* and the *discrete* components of the power spectral density of $v_l(t)$.

3.4–2 Power Spectral Density of Linearly Modulated Signals

In linearly modulated signals, which include ASK, PSK, and QAM as special cases, the lowpass equivalent of the modulated signal is of the form

$$v_l(t)=\sum_{n=-\infty}^{\infty}I_n g(t-nT) \qquad (3.4\text{–}13)$$

where $\{I_n\}$ is the stationary information sequence and $g(t)$ is the basic modulation pulse. Comparing Equations 3.4–13 and 3.4–1, we have

$$s_l(t,\boldsymbol{I}_n)=I_n g(t) \qquad (3.4\text{–}14)$$

from which

$$\begin{aligned}
G_k(f) &= \mathrm{E}\left[S_l(f;\boldsymbol{I}_k)S_l^*(f;\boldsymbol{I}_0)\right]\\
&= \mathrm{E}\left[I_k I_0^*|G(f)|^2\right]\\
&= R_I(k)|G(f)|^2
\end{aligned} \qquad (3.4\text{–}15)$$

where $R_I(k)$ represents the autocorrelation function of the information sequence $\{I_n\}$, and $G(f)$ is the Fourier transform of $g(t)$. Using Equation 3.4–15 in Equation 3.4–6 yields

$$\begin{aligned}\mathcal{S}_{v_l}(f) &= \frac{1}{T}|G(f)|^2 \sum_{k=-\infty}^{\infty} R_I(k)e^{-j2\pi kfT} \\ &= \frac{1}{T}|G(f)|^2 \mathcal{S}_I(f)\end{aligned} \tag{3.4–16}$$

where

$$\mathcal{S}_I(f) = \sum_{k=-\infty}^{\infty} R_I(k)e^{-j2\pi kfT} \tag{3.4–17}$$

represents the power spectral density of the discrete-time random process $\{I_n\}$.

Note that two factors determine the shape of the power spectral density as given in Equation 3.4–16. The first factor is the shape of the basic pulse used for modulation. The shape of this pulse obviously has an important impact on the power spectral density of the modulated signal. Smoother pulses result in more compact power spectral densities. Another factor that affects the power spectral density of the modulated signal is the power spectral density of the information sequence $\{I_n\}$ which is determined by the correlation properties of the information sequence. One method to control the power spectral density of the modulated signal is through controlling the correlation properties of the information sequence by passing it through an invertible linear filter prior to modulation. This linear filter controls the correlation properties of the modulated signals, and since it is invertible, the original information sequence can be retrieved from it. This technique is called *spectral shaping by precoding*.

For instance, we can employ a precoding of the form $J_n = I_n + \alpha I_{n-1}$, and by changing the value of α, we can control the power spectral density of the resulting modulated waveform. In general, we can introduce a memory of length L and define a precoding of the form

$$J_n = \sum_{k=0}^{L} \alpha_k I_{n-k} \tag{3.4–18}$$

and then generate the modulated waveform

$$v_l(t) = \sum_{k=-\infty}^{\infty} J_k\, g(t-kT) \tag{3.4–19}$$

Since the precoding operation is a linear operation, the resulting power spectral density is of the form

$$\mathcal{S}_{v_l}(f) = \frac{1}{T}|G(f)|^2 \left|\sum_{k=0}^{L} \alpha_k e^{-j2\pi kfT}\right|^2 \mathcal{S}_I(f) \tag{3.4–20}$$

Changing α_k's controls the power spectral density.

EXAMPLE 3.4–1. In a binary communication system $I_n = \pm 1$ with equal probability, and the I_n's are independent. This information stream linearly modulates a basic pulse of the form

$$g(t) = \Pi\left(\frac{t}{T}\right)$$

to generate

$$v(t) = \sum_{k=-\infty}^{\infty} I_k g(t - kT)$$

The power spectral density of the modulated signal will be of the form

$$\mathcal{S}_v(f) = \frac{1}{T}|T\operatorname{sinc}(Tf)|^2 \mathcal{S}_I(f)$$

To determine $\mathcal{S}_I(f)$, we need to find $R_I(k) = \mathrm{E}\left[I_{n+k}I_n^*\right]$. By independence of the $\{I_n\}$ sequence we have

$$R_I(k) = \begin{cases} \mathrm{E}\left[|I|^2\right] = 1 & k = 0 \\ \mathrm{E}\left[I_{n+k}\right]\mathrm{E}\left[I_n^*\right] = 0 & k \neq 0 \end{cases}$$

and hence

$$\mathcal{S}_I(f) = \sum_{k=-\infty}^{\infty} R_I(k) e^{-j2\pi kfT} = 1$$

Thus,

$$\mathcal{S}_v(f) = T\operatorname{sinc}^2(\tau f)$$

A precoding of the form

$$J_n = I_n + \alpha I_{n-1}$$

where α is real would result in a power spectral density of the form

$$\mathcal{S}_v(f) = T\operatorname{sinc}^2(Tf)\left|1 + \alpha e^{-j2\pi fT}\right|^2$$

or

$$\mathcal{S}_v(f) = T\operatorname{sinc}^2(Tf)\left(1 + \alpha^2 + 2\alpha\cos(2\pi fT)\right)$$

Choosing $\alpha = 1$ would result in a power spectral density that has a null at frequency $f = \frac{1}{2T}$. Note that this spectral null is independent of the shape of the basic pulse $g(t)$; that is, any other $g(t)$ having a precoding of the form $J_n = I_n + I_{n-1}$ will result in a spectral null at $f = \frac{1}{2T}$.

3.4–3 Power Spectral Density of Digitally Modulated Signals with Finite Memory

We now focus on a special case where the data sequence $\{I_n\}$ is such that I_n and I_{n+k} are independent for $|k| > K$, where K is a positive integer representing the memory in the information sequence. With this assumption, $S_l(f; \boldsymbol{I}_k)$ and $S_l^*(f; \boldsymbol{I}_0)$ are independent for $k > K$, and by stationarity have equal expected values. Therefore,

$$G_k(f) = |\mathrm{E}\left[S_l(f; \boldsymbol{I}_0)\right]|^2 = G_{K+1}(f), \quad \text{for } |k| > K \tag{3.4–21}$$

Obviously, $G_{K+1}(f)$ is real. Let us define

$$G''_k(f) = G_k(f) - G_{K+1}(f) = G_k(f) - |\mathrm{E}[S_l(f;\, \boldsymbol{I}_0)]|^2 \tag{3.4–22}$$

It is clear that $G''_k(f) = 0$ for $|k| > K$ and $G''_{-k}(f) = G''^{*}_k(f)$. Also note that

$$G''_0(f) = G_0(f) - G_{K+1}(f) = \mathrm{E}\left[|S_l(f;\, \boldsymbol{I}_0)|^2\right] - |\mathrm{E}[S_l(f;\, \boldsymbol{I}_0)]|^2 = \mathrm{VAR}[S_l(f;\, \boldsymbol{I}_0)] \tag{3.4–23}$$

In this case we can write Equation 3.4–6 in the following form:

$$\begin{aligned}
\mathcal{S}_{v_l}(f) &= \frac{1}{T}\sum_{k=-\infty}^{\infty}(G_k(f) - G_{K+1}(f))\, e^{-j2\pi kfT} + \frac{1}{T}\sum_{k=-\infty}^{\infty} G_{K+1}(f)\, e^{-j2\pi kfT} \\
&= \frac{1}{T}\sum_{k=-K}^{K} G''_k(f)\, e^{-j2\pi kfT} + \frac{1}{T^2}\sum_{k=-\infty}^{\infty} G_{K+1}(f)\,\delta\left(f - \frac{k}{T}\right) \\
&= \frac{1}{T}\mathrm{VAR}[S_l(f;\, \boldsymbol{I}_0)] + \frac{2}{T}\mathrm{Re}\left[\sum_{k=1}^{K} G''_k(f)\, e^{-j2\pi kfT}\right] \\
&\quad + \frac{1}{T^2}\sum_{k=-\infty}^{\infty} G_{K+1}\left(\frac{k}{T}\right)\delta\left(f - \frac{k}{T}\right) \\
&= \mathcal{S}^{(c)}_{v_l}(f) + \mathcal{S}^{(d)}_{v_l}(f)
\end{aligned} \tag{3.4–24}$$

The continuous and discrete components of the power spectral density in this case can be expressed as

$$\begin{aligned}
\mathcal{S}^{(c)}_{v_l}(f) &= \frac{1}{T}\mathrm{VAR}[S_l(f;\, \boldsymbol{I}_0)] + \frac{2}{T}\mathrm{Re}\left[\sum_{k=1}^{K} G''_k(f)\, e^{-j2\pi kfT}\right] \\
\mathcal{S}^{(d)}_{v_l}(f) &= \frac{1}{T^2}\sum_{k=-\infty}^{\infty} G_{K+1}\left(\frac{k}{T}\right)\delta\left(f - \frac{k}{T}\right)
\end{aligned} \tag{3.4–25}$$

Note that if $G_{K+1}\left(\frac{k}{T}\right) = 0$ for $k = 0, \pm 1, \pm 2, \ldots$, the discrete component of the power spectrum vanishes. Since $G_{K+1}(f) = |\mathrm{E}[S_l(f;\, \boldsymbol{I}_0)]|^2$, having $\mathrm{E}[s_l(t;\, \boldsymbol{I}_0)] = 0$ guarantees a continuous power spectral density with no discrete components.

3.4–4 Power Spectral Density of Modulation Schemes with a Markov Structure

The power spectral density of modulation schemes with memory was derived in Equations 3.4–6, 3.4–7, and 3.4–10. These results can be generalized to the general class of modulation systems that can be described in terms of a Markov chain. This is done by defining

$$\boldsymbol{I}_n = (S_{n-1}, I_n) \tag{3.4–26}$$

where $S_{n-1} \in (1, 2, \ldots, K)$ denotes the state of the modulator at time $n-1$ and I_n is the nth output of the information source. With the assumption that the Markov chain is homogeneous, the source is stationary, and the Markov chain has achieved its steady-state probabilities, the results of Section 3.4–1 apply and the power spectral density can be derived.

In the particular case where the signals generated by the modulator are determined by the state of the Markov chain, the derivation becomes simpler. Let us assume that the Markov chain that determines signal generation has a probability transition matrix denoted by $\boldsymbol{P}$. Let us further assume that the number of states is K and the signal generated when the modulator is in state i, $1 \leq i \leq K$, is denoted by $s_{il}(t)$. The steady-state probabilities of the states of the Markov chain are denoted by p_i, $1 \leq i \leq K$, and elements of the matrix $\boldsymbol{P}$ are denoted by P_{ij}, $1 \leq i, j \leq K$. With these assumptions the results of Section 3.4–1 can be applied, and the power spectral density may be expressed in the general form (see Tausworth and Welch, 1961)

$$\begin{aligned} \mathcal{S}_v(f) = \frac{1}{T^2} \sum_{n=-\infty}^{\infty} \left| \sum_{i=1}^{K} p_i S_{il}\left(\frac{n}{T}\right) \right|^2 \delta\left(f - \frac{n}{T}\right) + \frac{1}{T} \sum_{i=1}^{K} p_i |S'_{il}(f)|^2 \\ + \frac{2}{T} \operatorname{Re}\left[\sum_{i=1}^{K}\sum_{j=1}^{K} p_i S'^{*}_{il}(f) S'_{jl}(f) P_{ij}(f)\right] \end{aligned} \tag{3.4–27}$$

where $S_{il}(f)$ is the Fourier transform of the signal waveform $s_{il}(t)$ and

$$s'_{il}(t) = s_{il}(t) - \sum_{k=1}^{K} p_k s_{kl}(t) \tag{3.4–28}$$

$P_{ij}(f)$ is the Fourier transform of n-step state transition probabilities $P_{ij}(n)$, defined as

$$P_{ij}(f) = \sum_{n=1}^{\infty} P_{ij}(n) e^{-j2\pi n f T} \tag{3.4–29}$$

and K is the number of states of the modulator. The term $P_{ij}(n)$ denotes the probability that the signal $s_j(t)$ is transmitted n signaling intervals after the transmission of $s_i(t)$. Hence, $\{P_{ij}(n)\}$ are the transition probabilities in the transition probability matrix $\boldsymbol{P}^n$. Note that $P_{ij}(1) = P_{ij}$, the (i, j)th entry in $\boldsymbol{P}$.

When there is no memory in the modulation method, the signal waveform transmitted on each signaling interval is independent of the waveforms transmitted in previous signaling intervals. The power density spectrum of the resultant signal may still be expressed in the form of Equation 3.4–27, if the transition probability matrix is replaced by

$$\boldsymbol{P} = \begin{bmatrix} p_1 & p_2 & \cdots & p_K \\ p_1 & p_2 & \cdots & p_K \\ \vdots & \vdots & \ddots & \vdots \\ p_1 & p_2 & \cdots & p_K \end{bmatrix} \tag{3.4–30}$$

and we impose the condition that $\boldsymbol{P}^n = \boldsymbol{P}$ for all $n \geq 1$. Under these conditions, the expression for the power density spectrum becomes a function of the stationary state probabilities $\{p_i\}$ only, and hence it reduces to the simpler form

$$\begin{aligned}\mathcal{S}_{v_l}(f) &= \frac{1}{T^2}\sum_{n=-\infty}^{\infty}\left|\sum_{i=1}^{K} p_i S_{il}\left(\frac{n}{T}\right)\right|^2 \delta\left(f-\frac{n}{T}\right)\\ &+\frac{1}{T}\sum_{i=1}^{K} p_i(1-p_i)|S_{il}(f)|^2\\ &-\frac{2}{T}\sum_{i=1}^{K}\sum_{\substack{j=1\\ i<j}}^{K} p_i p_j \operatorname{Re}\left[S_{il}(f)S_{jl}^*(f)\right]\end{aligned} \tag{3.4–31}$$

We observe that when

$$\sum_{i=1}^{K} p_i S_{il}\left(\frac{n}{T}\right) = 0 \tag{3.4–32}$$

the discrete component of the power spectral density in Equation 3.4–31 vanishes. This condition is usually imposed in the design of digital communication systems and is easily satisfied by an appropriate choice of signaling waveforms (Problem 3.34).

EXAMPLE 3.4–2. Let us determine the power density spectrum of the baseband-modulated NRZ signal described in Section 3.3. The NRZ signal is characterized by the two waveforms $s_1(t) = g(t)$ and $s_2(t) = -g(t)$, where $g(t)$ is a rectangular pulse of amplitude A. For $K = 2$, Equation 3.4–31 reduces to

$$\mathcal{S}_v(f) = \frac{(2p-1)^2}{T^2}\sum_{n=-\infty}^{\infty}\left|G\left(\frac{n}{T}\right)\right|^2\delta\left(f-\frac{n}{T}\right) + \frac{4p(1-p)}{T}|G(f)|^2 \tag{3.4–33}$$

where

$$|G(f)|^2 = (AT)^2\operatorname{sinc}^2(fT)$$

Observe that when $p = \frac{1}{2}$, the line spectrum vanishes and $\mathcal{S}_v(f)$ reduces to

$$\mathcal{S}_v(f) = \frac{1}{T}|G(f)|^2 \tag{3.4–34}$$

EXAMPLE 3.4–3. The NRZI signal is characterized by the transition probability matrix

$$\boldsymbol{P} = \begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ \frac{1}{2} & \frac{1}{2}\end{bmatrix}$$

Notice that in this case $\boldsymbol{P}^n = \boldsymbol{P}$ for all $n \geq 1$. Hence, the special form for the power density spectrum given by Equation 3.4–33 applies to this modulation format as well. Consequently, the power density spectrum for the NRZI signal is identical to the spectrum of the NRZ signal.

3.4–5 Power Spectral Densities of CPFSK and CPM Signals

In this section, we derive the power density spectrum for the class of constant-amplitude CPM signals described in Sections 3.3–1 and 3.3–2. We begin by computing the autocorrelation function and its Fourier transform.

The constant-amplitude CPM signal is expressed as

$$s(t;\boldsymbol{I}) = A\cos[2\pi f_c t + \phi(t;\boldsymbol{I})] \tag{3.4–35}$$

where

$$\phi(t;\boldsymbol{I}) = 2\pi h \sum_{k=-\infty}^{\infty} I_k q(t-kT) \tag{3.4–36}$$

Each symbol in the sequence $\{I_n\}$ can take one of the M values $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$. These symbols are statistically independent and identically distributed with prior probabilities

$$P_n = P(I_k = n), \qquad n = \pm 1, \pm 3, \ldots, \pm(M-1) \tag{3.4–37}$$

where $\sum_n P_n = 1$. The pulse $g(t) = q'(t)$ is zero outside of the interval $[0, LT]$, $q(t) = 0, t < 0$, and $q(t) = \frac{1}{2}$ for $t > LT$.

The autocorrelation function of the equivalent lowpass signal

$$v_l(t) = e^{j\phi(t,\boldsymbol{I})} \tag{3.4–38}$$

is

$$R_{v_l}(t+\tau;t) = \mathrm{E}\left[\exp\left(j2\pi h \sum_{k=-\infty}^{\infty} I_k[q(t+\tau-kT) - q(t-kT)]\right)\right] \tag{3.4–39}$$

First, we express the sum in the exponent as a product of exponents. The result is

$$R_{v_l}(t+\tau;t) = \mathrm{E}\left[\prod_{k=-\infty}^{\infty} \exp\{j2\pi h I_k[q(t+\tau-kT) - q(t-kT)]\}\right] \tag{3.4–40}$$

Next, we perform the expectation over the data symbols $\{I_k\}$. Since these symbols are statistically independent, we obtain

$$R_{v_l}(t+\tau;t) = \prod_{k=-\infty}^{\infty}\left(\sum_{\substack{n=-(M-1)\\ n\ \text{odd}}}^{M-1} P_n \exp\{j2\pi h n[q(t+\tau-kT) - q(t-kT)]\}\right) \tag{3.4–41}$$

Finally, the average autocorrelation function is

$$\bar{R}_{v_l}(\tau) = \frac{1}{T}\int_0^{T_0} R_{v_l}(t+\tau;t)\,dt \tag{3.4–42}$$

Although Equation 3.4–41 implies that there are an infinite number of factors in the product, the pulse $g(t) = q'(t) = 0$ for $t < 0$ and $t > LT$, and $q(t) = 0$ for $t < 0$. Consequently only a finite number of terms in the product have nonzero exponents. Thus Equation 3.4–41 can be simplified considerably. In addition, if we let $\tau = \xi + mT$, where $0 \le \xi < T$ and $m = 0, 1, \ldots$, the average autocorrelation in Equation 3.4–42 reduces to

$$\bar{R}_{v_l}(\xi + mT) = \frac{1}{T}\int_0^{T}\prod_{k=1-L}^{m+1}\left(\sum_{\substack{n=-(M-1)\\ n\ \text{odd}}}^{M-1} P_n e^{j2\pi h n[q(t+\xi-(k-m)T) - q(t-kT)]}\right)dt \tag{3.4–43}$$

Let us focus on $\bar{R}_{v_l}(\xi + mT)$ for $\xi + mT \geq LT$. In this case, Equation 3.4–43 may be expressed as

$$\bar{R}_{v_l}(\xi + mT) = [\Phi_I(h)]^{m-L} \lambda(\xi), \qquad m \geq L, \qquad 0 \leq \xi < T \tag{3.4–44}$$

where $\Phi_I(h)$ is the characteristic function of the random sequence $\{I_n\}$, defined as

$$\begin{aligned}\Phi_I(h) &= \mathrm{E}\,[e^{j\pi h I_n}] \\ &= \sum_{\substack{n=-(M-1)\\ n \text{ odd}}}^{M-1} P_n e^{j\pi hn}\end{aligned} \tag{3.4–45}$$

and $\lambda(\xi)$ is the remaining part of the average autocorrelation function, which may be expressed as

$$\begin{aligned}\lambda(\xi) = \frac{1}{T}\int_0^T \prod_{k=1-L}^{0} &\left(\sum_{\substack{n=-(M-1)\\ n \text{ odd}}}^{M-1} P_n \exp\left\{ j2\pi hn \left[\frac{1}{2} - q(t-kT) \right] \right\} \right) \\ \times \prod_{k=m-L}^{m+1} &\left(\sum_{\substack{n=-(M-1)\\ n \text{ odd}}}^{M-1} P_n \exp[j2\pi hnq(t+\xi-kT)] \right) dt, \qquad m \geq L\end{aligned} \tag{3.4–46}$$

Thus, $\bar{R}_{v_l}(\tau)$ may be separated into a product of $\lambda(\xi)$ and $\Phi_I(h)$ as indicated in Equation 3.4–44 for $\tau = \xi + mT \geq LT$ and $0 \leq \xi < T$. This property is used below.

The Fourier transform of $\bar{R}_{v_l}(\tau)$ yields the average power density spectrum as

$$\begin{aligned}\mathcal{S}_{v_l}(f) &= \int_{-\infty}^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, dt \\ &= 2\,\mathrm{Re}\left[\int_0^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau \right]\end{aligned} \tag{3.4–47}$$

But

$$\begin{aligned}\int_0^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau &= \int_0^{LT} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau \\ &\quad + \int_{LT}^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau\end{aligned} \tag{3.4–48}$$

With the aid of Equation 3.4–44, the integral in the range $LT \leq \tau < \infty$ may be expressed as

$$\int_{LT}^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau = \sum_{m=L}^{\infty} \int_{mT}^{(m+1)T} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau \tag{3.4–49}$$

Now, let $\tau = \xi + mT$. Then Equation 3.4–49 becomes

$$\begin{aligned}\int_{LT}^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, dt &= \sum_{m=L}^{\infty} \int_0^T \bar{R}_{v_l}(\xi + mT) e^{-j2\pi f(\xi+mT)}\, d\xi \\ &= \sum_{m=L}^{\infty} \int_0^T \lambda(\xi)[\Phi_I(h)]^{m-L} e^{-j2\pi f(\xi+mT)}\, d\xi \qquad (3.4\text{–}50) \\ &= \sum_{n=0}^{\infty} \Phi_I^n(h) e^{-j2\pi f nT} \int_0^T \lambda(\xi) e^{-j2\pi f(\xi+LT)} d\xi\end{aligned}$$

A property of the characteristic function is $|\Phi_I(h)| \le 1$. For values of h for which $|\Phi_I(h)| < 1$, the summation in Equation 3.4–50 converges and yields

$$\sum_{n=0}^{\infty} \Phi_I^n(h) e^{-j2\pi f nT} = \frac{1}{1 - \Phi_I(h) e^{-j2\pi fT}} \qquad (3.4\text{–}51)$$

In this case, Equation 3.4–50 reduces to

$$\int_{LT}^{\infty} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, dt = \frac{1}{1 - \Phi_I(h) e^{-j2\pi fT}} \int_0^T \bar{R}_{v_l}(\xi + LT) e^{-j2\pi f(\xi+LT)}\, d\xi \qquad (3.4\text{–}52)$$

By combining Equations 3.4–47, 3.4–48, and 3.4–52, we obtain the power density spectrum of the CPM signal in the form

$$\mathcal{S}_{v_l}(f) = 2\,\mathrm{Re}\left[\int_0^{LT} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau + \frac{1}{1 - \Phi_I(h) e^{-j2\pi fT}} \int_{LT}^{(L+1)T} \bar{R}_{v_l}(\tau) e^{-j2\pi f\tau}\, d\tau\right] \qquad (3.4\text{–}53)$$

This is the desired result when $|\Phi_I(h)| < 1$. In general, the power density spectrum is evaluated numerically from Equation 3.4–53. The average autocorrelation function $\bar{R}_{v_l}(\tau)$ for the range $0 \le \tau \le (L+1)T$ may be computed numerically from Equation 3.4–43.

For values of h for which $|\Phi_I(h)| = 1$, e.g., $h = K$, where K is an integer, we can set

$$\Phi_I(h) = e^{j2\pi\nu}, \qquad 0 \le \nu < 1 \qquad (3.4\text{–}54)$$

Then the sum in Equation 3.4–50 becomes

$$\sum_{n=0}^{\infty} e^{-j2\pi T(f-\nu/T)n} = \frac{1}{2} + \frac{1}{2T} \sum_{n=-\infty}^{\infty} \delta\left(f - \frac{\nu}{T} - \frac{n}{T}\right) - j\frac{1}{2}\cot \pi T\left(f - \frac{\nu}{T}\right) \qquad (3.4\text{–}55)$$

Thus, the power density spectrum now contains impulses located at frequencies

$$f_n = \frac{n+\nu}{T}, \qquad 0 \le \nu < 1, \qquad n = 0, 1, 2, \ldots \qquad (3.4\text{–}56)$$

The result in Equation 3.4–55 can be combined with Equations 3.4–50 and 3.4–48 to obtain the entire power density spectrum, which includes both a continuous spectrum component and a discrete spectrum component.

Let us return to the case for which $|\Phi_I(h)| < 1$. When symbols are equally probable, i.e.,

$$P_n = \frac{1}{M} \qquad \text{for all } n \tag{3.4–57}$$

the characteristic function simplifies to the form

$$\begin{aligned}\Phi_I(h) &= \frac{1}{M} \sum_{\substack{n=-(M-1)\\ n\text{odd}}}^{M-1} e^{j\pi h n} \\ &= \frac{1}{M} \frac{\sin M\pi h}{\sin \pi h}\end{aligned} \tag{3.4–58}$$

Note that in this case $\Phi_I(h)$ is real. The average autocorrelation function given by Equation 3.4–43 also simplifies in this case to

$$\bar{R}_{v_l}(\tau) = \frac{1}{2T} \int_0^T \prod_{k=1-L}^{[\tau/T]} \frac{1}{M} \frac{\sin 2\pi h M[q(t+\tau-kT) - q(t-kT)]}{\sin 2\pi h[q(t+\tau-kT) - q(t-kT)]} dt \tag{3.4–59}$$

The corresponding expression for the power density spectrum reduces to

$$\begin{aligned}\mathcal{S}_{v_l}(f) = 2 \Bigg[&\int_0^{LT} \bar{R}_{v_l}(\tau) \cos 2\pi f \tau \, d\tau \\ &+ \frac{1 - \Phi_I(h)\cos 2\pi f T}{1 + \Phi_I^2(h) - 2\Phi_I(h)\cos 2\pi f T} \int_{LT}^{(L+1)T} \bar{R}_{v_l}(\tau)\cos 2\pi f\tau \, d\tau \\ &- \frac{\Phi_I(h)\sin 2\pi f T}{1 + \Phi_I^2(h) - 2\Phi_I(h)\cos 2\pi f T} \int_{LT}^{(L+1)T} \bar{R}_{v_l}(\tau)\sin 2\pi f\tau \, d\tau \Bigg]\end{aligned} \tag{3.4–60}$$

Power Spectral Density of CPFSK

A closed-form expression for the power density spectrum can be obtained from Equation 3.4–60 when the pulse shape $g(t)$ is rectangular and zero outside the interval $[0, T]$. In this case, $q(t)$ is linear for $0 \le t \le T$. The resulting power spectrum may be expressed as

$$\mathcal{S}_v(f) = T \left[\frac{1}{M} \sum_{n=1}^{M} A_n^2(f) + \frac{2}{M^2} \sum_{n=1}^{M} \sum_{m=1}^{M} B_{nm}(f) A_n(f) A_m(f) \right] \tag{3.4–61}$$

where

$$\begin{aligned} A_n(f) &= \frac{\sin \pi[fT - \frac{1}{2}(2n-1-M)h]}{\pi[fT - \frac{1}{2}(2n-1-M)h]} \\ B_{nm}(f) &= \frac{\cos(2\pi f T - \alpha_{nm}) - \Phi \cos \alpha_{nm}}{1 + \Phi^2 - 2\Phi\cos 2\pi f T} \\ \alpha_{nm} &= \pi h(m+n-1-M) \\ \Phi &\equiv \Phi(h) = \frac{\sin M\pi h}{M \sin \pi h} \end{aligned} \tag{3.4–62}$$

The power density spectrum of CPFSK for $M = 2, 4$, and 8 is plotted in Figures 3.4–1 to 3.4–3 as a function of the normalized frequency fT, with the modulation index $h = 2f_dT$ as a parameter. Note that only one-half of the bandwidth occupancy is shown in these graphs. The origin corresponds to the carrier f_c. The graphs illustrate that the spectrum of CPFSK is relatively smooth and well confined for $h < 1$. As h approaches unity, the spectra become very peaked, and for $h = 1$ when $|\Phi| = 1$, we find that impulses occur at M frequencies. When $h > 1$, the spectrum becomes much

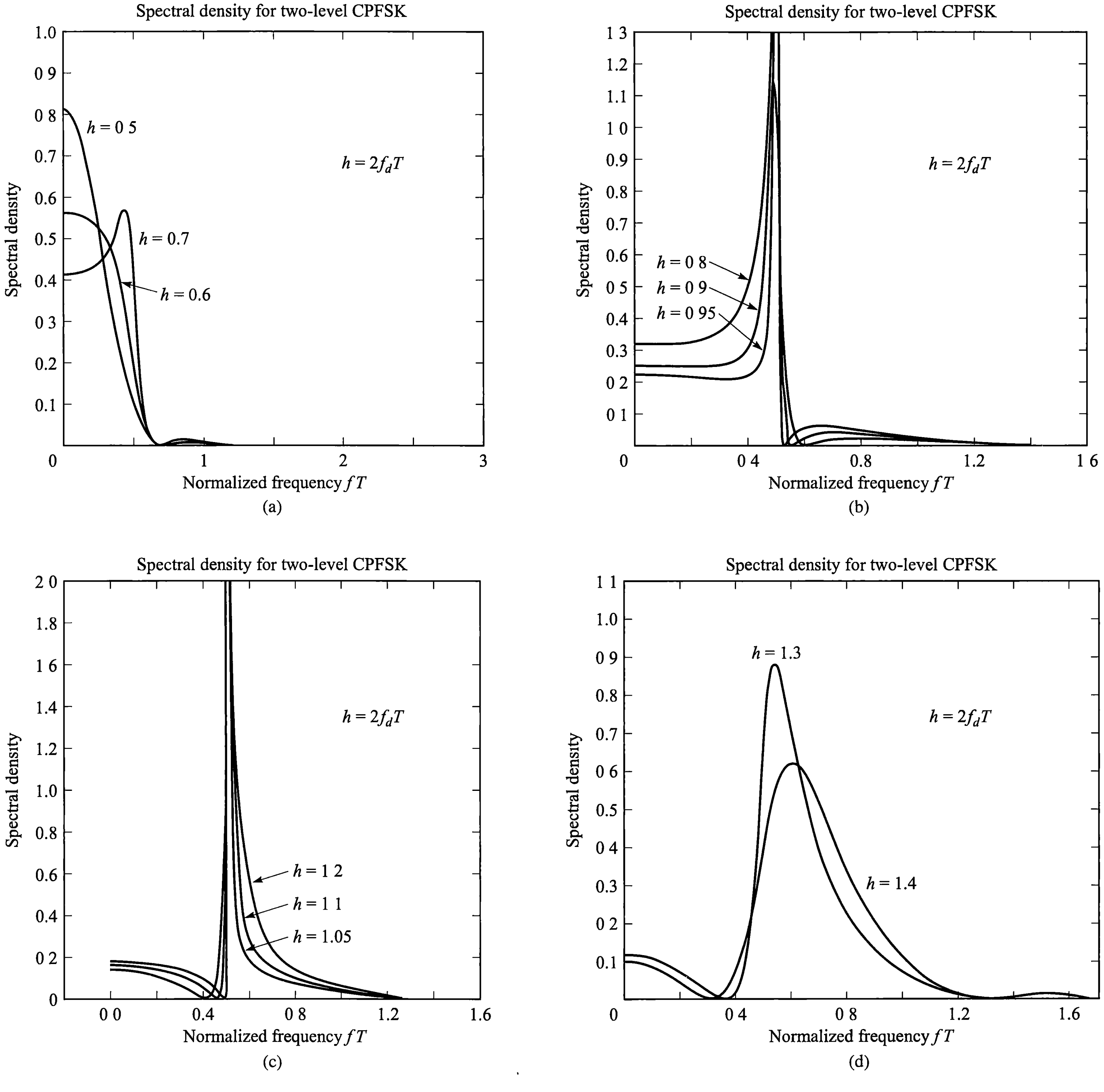
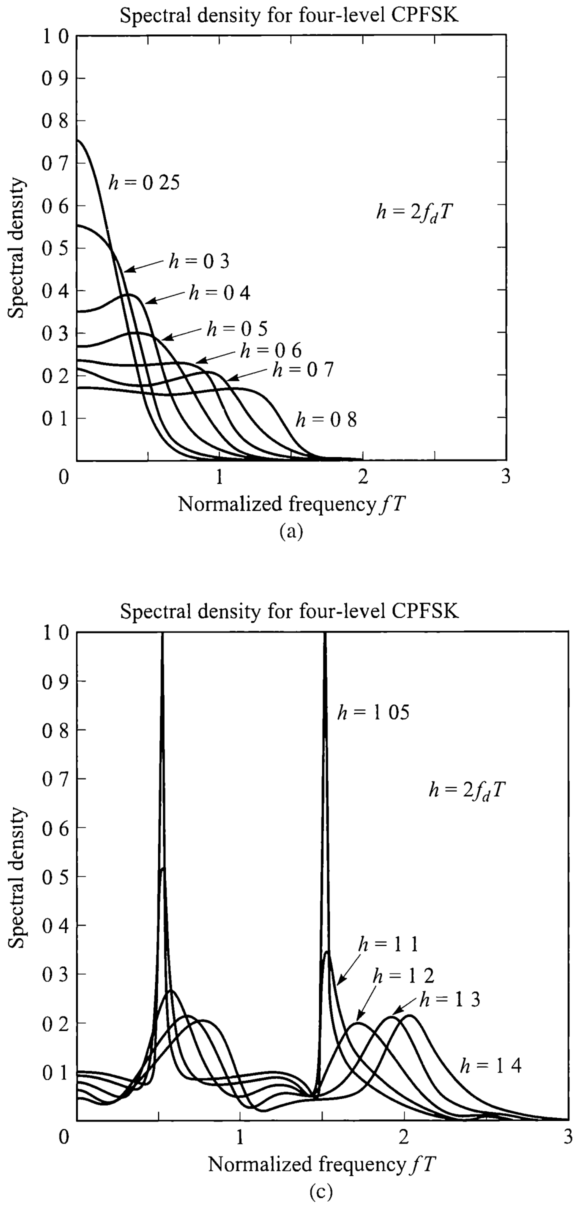


FIGURE 3.4–1
Power spectral density of binary CPFSK.



(a) (b) (c)

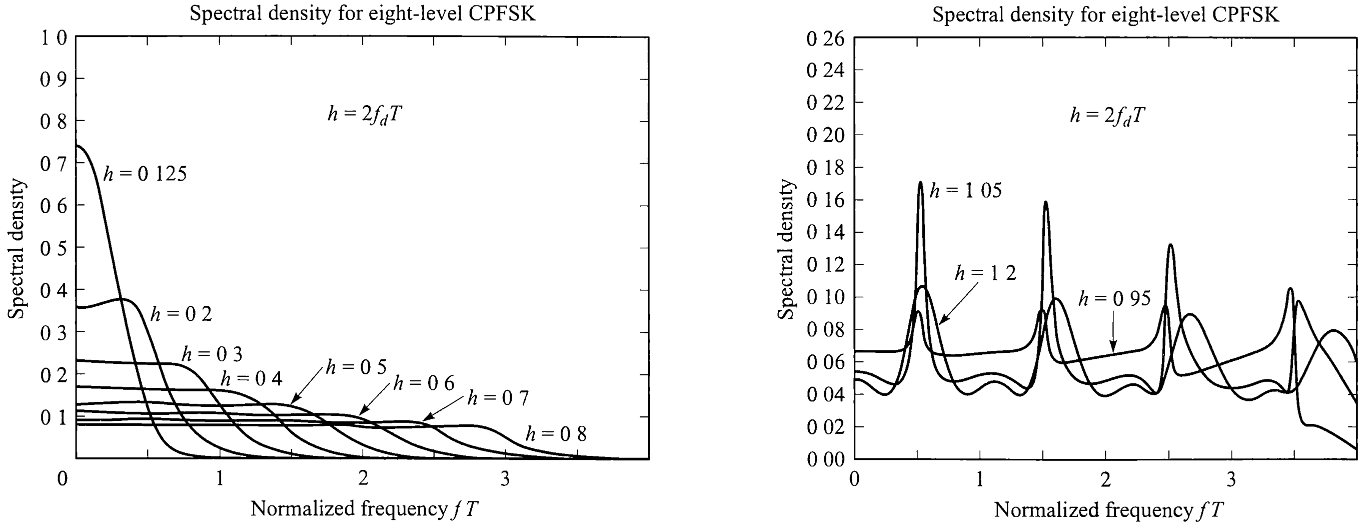
FIGURE 3.4–2
Power spectral density of quaternary CPFSK.

broader. In communication systems where CPFSK is used, the modulation index is designed to conserve bandwidth, so that $h < 1$.

The special case of binary CPFSK with $h = \frac{1}{2}$ (or $f_d = 1/4T$) and $\Phi = 0$ corresponds to MSK. In this case, the spectrum of the signal is

$$\mathcal{S}_v(f) = \frac{16A^2T}{\pi^2}\left(\frac{\cos 2\pi fT}{1-16f^2T^2}\right)^2 \tag{3.4–63}$$

where the signal amplitude $A = 1$ in Equation 3.4–62. In contrast, the spectrum of four-phase offset (quadrature) PSK (OQPSK) with a rectangular pulse $g(t)$ of



(a) (b)

FIGURE 3.4–3
Power spectral density of octal CPFSK.

duration T is

$$\mathcal{S}_v(f) = A^2 T \left(\frac{\sin \pi f T}{\pi f T} \right)^2 \tag{3.4–64}$$

If we compare these spectral characteristics, we should normalize the frequency variable by the bit rate or the bit interval T_b. Since MSK is binary FSK, it follows that $T = T_b$ in Equation 3.4–63. On the other hand, in OQPSK, $T = 2T_b$ so that Equation 3.4–64 becomes

$$\mathcal{S}_v(f) = 2A^2 T_b \left(\frac{\sin 2\pi f T_b}{2\pi f T_b} \right)^2 \tag{3.4–65}$$

The spectra of the MSK and OQPSK signals are illustrated in Figure 3.4–4. Note that the main lobe of MSK is 50 percent wider than that for OQPSK. However, the side lobes in MSK fall off considerably faster. For example, if we compare the bandwidth W that contains 99 percent of the total power, we find that $W = 1.2/T_b$ for MSK and $W \approx 8/T_b$ for OQPSK. Consequently, MSK has a narrower spectral occupancy when viewed in terms of fractional out-of-band power above $fT_b = 1$. Graphs for the fractional out-of-band power for OQPSK and MSK are shown in Figure 3.4–5. Note that MSK is significantly more bandwidth-efficient than QPSK. This efficiency accounts for the popularity of MSK in many digital communication systems.

Even greater bandwidth efficiency than MSK can be achieved by reducing the modulation index. However, the FSK signals will no longer be orthogonal, and there will be an increase in the error probability.

Spectral Characteristics of CPM

In general, the bandwidth occupancy of CPM depends on the choice of the modulation index h, the pulse shape $g(t)$, and the number of signals M. As we have observed

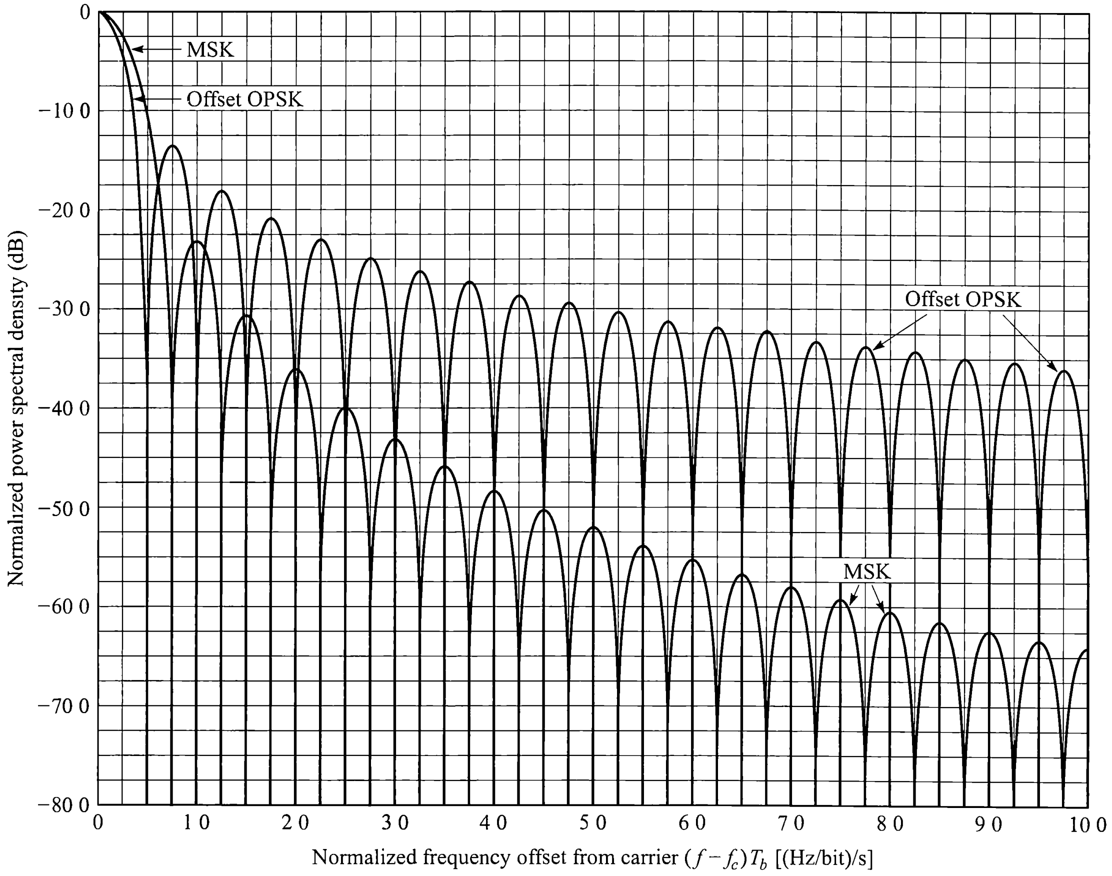


FIGURE 3.4–4
Power spectral density of MSK and OQPSK. [*Source:* Gronemeyer and McBride (1976); © IEEE.]

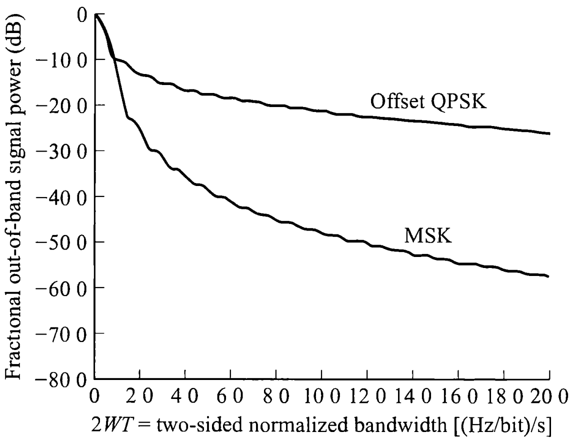


FIGURE 3.4–5
Fractional out-of-band power (normalized two-sided bandwidth = $2WT$). [*Source:* Gronemeyer and McBride (1976); © IEEE.]

for CPFSK, small values of h result in CPM signals with relatively small bandwidth occupancy, while large values of h result in signals with large bandwidth occupancy. This is also the case for the more general CPM signals.

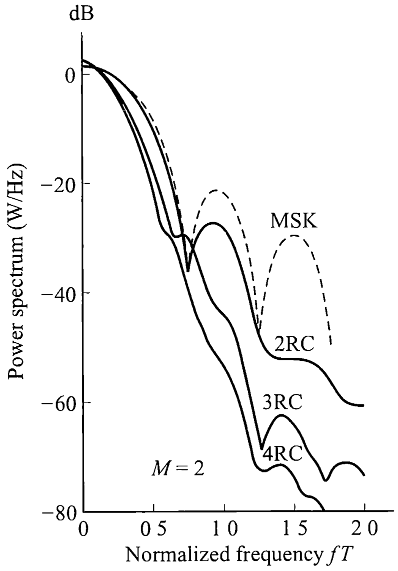


FIGURE 3.4–6
Power spectral density for binary CPM with $h = \frac{1}{2}$ and different pulse shapes. [*Source:* Aulin et al. (1981); © IEEE.]

The use of smooth pulses such as raised cosine pulses of the form

$$g(t) = \begin{cases} \frac{1}{2LT}\left(1 - \cos\frac{2\pi t}{LT}\right) & 0 \le t \le LT \\ 0 & \text{otherwise} \end{cases} \qquad (3.4\text{–}66)$$

where $L = 1$ for full response and $L > 1$ for partial response, results in smaller bandwidth occupancy and hence greater bandwidth efficiency than in the use of rectangular pulses. For example, Figure 3.4–6 illustrates the power density spectrum for binary CPM with different partial-response raised cosine (LRC) pulses when $h = \frac{1}{2}$. For comparison, the spectrum of binary CPFSK is also shown. Note that as L increases, the pulse $g(t)$ becomes smoother and the corresponding spectral occupancy of the signal is reduced.

The effect of varying the modulation index in a CPM signal is illustrated in Figure 3.4–7 for the case of $M = 4$ and a raised cosine pulse of the form given in Equation 3.4–66 with $L = 3$. Note that these spectral characteristics are similar to the

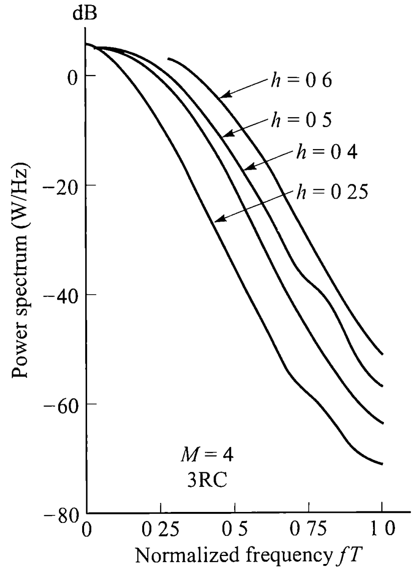


FIGURE 3.4–7
Power spectral density for $M = 4$ CPM with 3RC and different modulation indices. [*Source:* Aulin et al. (1981); © IEEE.]

ones illustrated previously for CPFSK, except that these spectra are narrower due to the use of a smoother pulse shape.

3.5
BIBLIOGRAPHICAL NOTES AND REFERENCES

The digital modulation methods introduced in this chapter are widely used in digital communication systems. Chapter 4 is concerned with optimum demodulation techniques for these signals and their performance in an additive white Gaussian noise channel. A general reference for signal characterization is the book by Franks (1969).

Of particular importance in the design of digital communication systems are the spectral characteristics of the digitally modulated signals, which are presented in this chapter in some depth. Of these modulation techniques, CPM is one of the most important due to its efficient use of bandwidth. For this reason, it has been widely investigated by many researchers, and a large number of papers have been published in the technical literature. The most comprehensive treatment of CPM, including its performance and its spectral characteristics, can be found in the book by Anderson et al. (1986). In addition to this text, the tutorial paper by Sundberg (1986) presents the basic concepts and an overview of the performance characteristics of various CPM techniques.

The linear representation of CPM was developed by Laurent (1986) for binary modulation. It was extended to M-ary CPM signals by Mengali and Morelli (1995). Rimoldi (1988) showed that a CPM system can be decomposed into a continuous-phase and a memoryless modulator. This paper also contains over 100 references to published papers on this topic.

There are a large number of references dealing with the spectral characteristics of CPFSK and CPM. As a point of reference, we should mention that MSK was invented by Doelz and Heald in 1961. The early work on the power spectral density of CPFSK and CPM was done by Bennett and Rice (1963), Anderson and Salz (1965), and Bennett and Davey (1965). The book by Lucky et al. (1968) also contains a treatment of the spectral characteristics of CPFSK. Most of the recent work is referenced in the paper by Sundberg (1986). We should also cite the special issue on bandwidth-efficient modulation and coding published by the *IEEE Transactions on Communications* (March 1981), which contains several papers on the spectral characteristics and performance of CPM. The generalization of MSK to multiple amplitudes was investigated by Weber et al. (1978). The combination of multiple amplitudes with general CPM was proposed by Mulligan (1988) who investigated its spectral characteristics and its error probability performance in Gaussian noise with and without coding.

PROBLEMS

3.1 Using the identity

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

show that

$$1^2 + 3^2 + 5^2 + \cdots + (M-1)^2 = \frac{M(M^2-1)}{6}$$

and derive Equation 3.2–5.

3.2 Determine the signal space representation of the four signals $s_k(t)$, $k = 1, 2, 3, 4$, shown in Figure P3.2, by using as basis functions the orthonormal functions $\phi_1(t)$ and $\phi_2(t)$. Plot the signal space diagram, and show that this signal set is equivalent to that for a four-phase PSK signal.

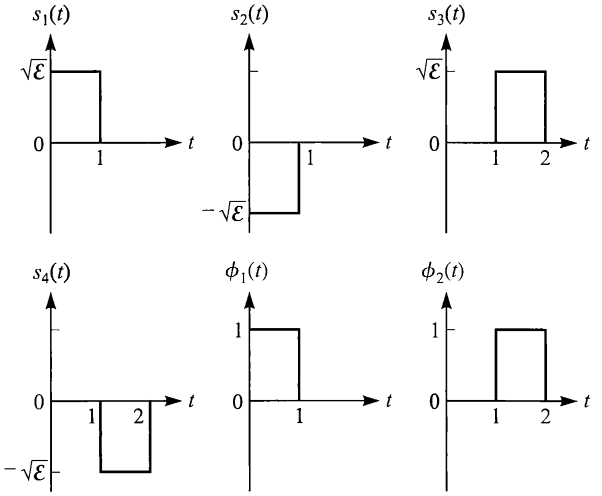

$s_1(t)$
$\sqrt{\mathcal{E}}$
0
1
t
$s_2(t)$
0
1
t
$-\sqrt{\mathcal{E}}$
$s_3(t)$
$\sqrt{\mathcal{E}}$
0
1
2
t
$s_4(t)$
0
1
2
t
$-\sqrt{\mathcal{E}}$
$\phi_1(t)$
1
0
1
t
$\phi_2(t)$
1
0
1
2
t

FIGURE P3.2

3.3 $\pi/4$-QPSK may be considered as two QPSK systems offset by $\pi/4$ rad.

1. Sketch the signal space diagram for a $\pi/4$-QPSK signal.
2. Using Gray encoding, label the signal points with the corresponding data bits.

3.4 Consider the octal signal point constellations in Figure P3.4.

1. The nearest-neighbor signal points in the 8-QAM signal constellation are separated in distance by A units. Determine the radii a and b of the inner and outer circles, respectively.
2. The adjacent signal points in the 8-PSK are separated by a distance of A units. Determine the radius r of the circle.

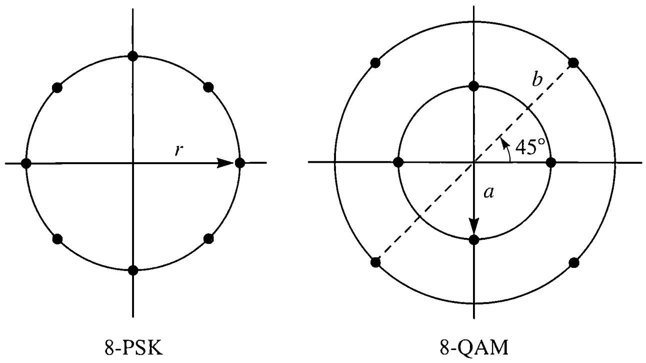

r
8-PSK
b
45°
a
8-QAM

FIGURE P3.4

3. Determine the average transmitter powers for the two signal constellations, and compare the two powers. What is the relative power advantage of one constellation over the other? (Assume that all signal points are equally probable.)

3.5 Consider the 8-point QAM signal constellation shown in Figure P3.4.

1. Is it possible to assign 3 data bits to each point of the signal constellation such that the nearest (adjacent) points differ in only 1 bit position?
2. Determine the symbol rate if the desired bit rate is 90 Mbits/s.

3.6 Consider the two 8-point QAM signal constellations shown in Figure P3.6. The minimum distance between adjacent points is $2A$. Determine the average transmitted power for each constellation, assuming that the signal points are equally probable. Which constellation is more power-efficient?

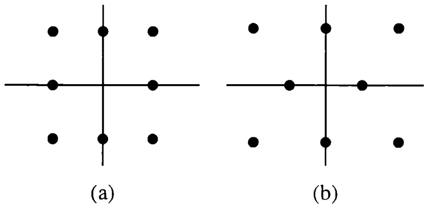


FIGURE P3.6

3.7 Specify a Gray code for the 16-QAM signal constellation shown in Figure P3.7.

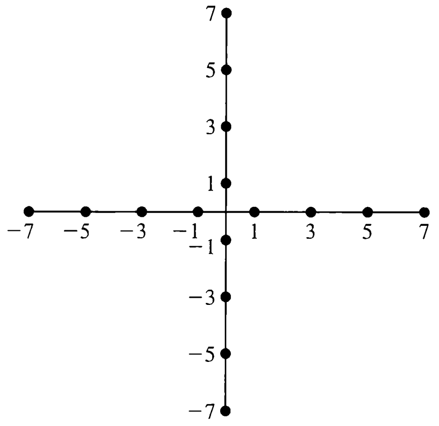


FIGURE P3.7

3.8 In an MSK signal, the initial state for the phase is either 0 or π rad. Determine the terminal phase state for the following four input pairs of input data:

1. 00
2. 01
3. 10
4. 11

3.9 Determine the number of states in the state trellis diagram for

1. A full-response binary CPFSK with $h = \frac{2}{3}$ or $\frac{3}{4}$.
2. A partial-response $L = 3$ binary CPFSK with $h = \frac{2}{3}$ or $\frac{3}{4}$.

3.10 A speech signal is sampled at a rate of 8 kHz, and then encoded using 8 bits per sample. The resulting binary data are then transmitted through an AWGN baseband channel via M-level PAM. Determine the bandwidth required for transmission when

1. $M = 4$
2. $M = 8$
3. $M = 16$

3.11 The power density spectrum of the cyclostationary process

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT)$$

can be derived by averaging the autocorrelation function $R_v(t + \tau, t)$ over the period T of the process and then evaluating the Fourier transform of the average autocorrelation function. An alternative approach is to change the cyclostationary process into a stationary process $v_\Delta(t)$ by adding a random variable Δ, uniformly distributed over $0 \le \Delta < T$, so that

$$v_\Delta(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT - \Delta)$$

and defining the spectral density of $v(t)$ as the Fourier transform of the autocorrelation function of the stationary process $v_\Delta(t)$. Derive the result in Equation 3.4–16 by evaluating the autocorrelation function of $v_\Delta(t)$ and its Fourier transform.

3.12 Show that 16-QAM can be represented as a superposition of two four-phase constant-envelope signals where each component is amplified separately before summing, i.e.,

$$s(t) = G(A_n \cos 2\pi f_c t + B_n \sin 2\pi f_c t) + (C_n \cos 2\pi f_c t + D_n \sin 2\pi f_c t)$$

where $\{A_n\}$, $\{B_n\}$, $\{C_n\}$, and $\{D_n\}$ are statistically independent binary sequences with elements from the set $\{+1, -1\}$ and G is the amplifier gain. Thus, show that the resulting signal is equivalent to

$$s(t) = I_n \cos 2\pi f_c t + Q_n \sin 2\pi f_c t$$

and determine I_n and Q_n in terms of A_n, B_n, C_n, and D_n.

3.13 Consider a four-phase PSK signal represented by the equivalent lowpass signal

$$u(t) = \sum_n I_n g(t - nT)$$

where I_n takes on one of the four possible values $\sqrt{\frac{1}{2}}(\pm 1 \pm j)$ with equal probability. The sequence of information symbols $\{I_n\}$ is statistically independent.

1. Determine and sketch the power density spectrum of $u(t)$ when

$$g(t) = \begin{cases} A & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases}$$

2. Repeat Part 1 when

$$g(t) = \begin{cases} A \sin(\pi t / T) & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases}$$

3. Compare the spectra obtained in Parts 1 and 2 in terms of the 3-dB bandwidth and the bandwidth to the first spectral zero.

3.14 A PAM partial-response signal (PRS) is generated as shown in Figure P3.14 by exciting an ideal lowpass filter of bandwidth W by the sequence

$$B_n = I_n + I_{n-1}$$

at a rate $1/T = 2W$ symbols/s. The sequence $\{I_n\}$ consists of binary digits selected independently from the alphabet $\{1, -1\}$ with equal probability. Hence, the filtered signal has the form

$$v(t) = \sum_{n=-\infty}^{\infty} B_n g(t - nT),\ T = \frac{1}{2W}$$

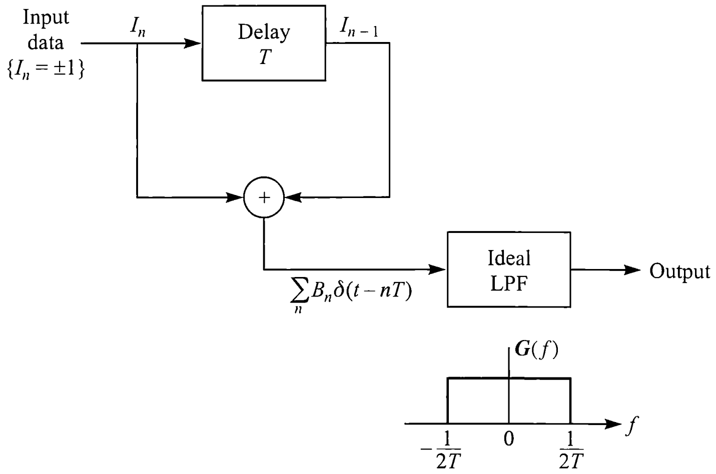


FIGURE P3.14

1. Sketch the signal space diagram for $v(t)$, and determine the probability of occurrence of each symbol.
2. Determine the autocorrelation and power density spectrum of the three-level sequence $\{B_n\}$.
3. The signal points of the sequence $\{B_n\}$ form a Markov chain. Sketch this Markov chain, and indicate the transition probabilities among the states.

3.15 The lowpass equivalent representation of a PAM signal is

$$u(t) = \sum_n I_n g(t - nT)$$

Suppose $g(t)$ is a rectangular pulse and

$$I_n = a_n - a_{n-2}$$

where $\{a_n\}$ is a sequence of uncorrelated binary-valued $(1, -1)$ random variables that occur with equal probability.

1. Determine the autocorrelation function of the sequence $\{I_n\}$.
2. Determine the power density spectrum of $u(t)$.
3. Repeat (2) if the possible values of the a_n are (0, 1).

3.16 Use the results in Section 3.4–4 to determine the power density spectrum of the binary FSK signals in which the waveforms are

$$s_i(t) = \sin \omega_i t, \qquad i = 1, 2, \qquad 0 \le t \le T$$

where $\omega_1 = n\pi/T$ and $\omega_2 = m\pi/T$, $n \neq m$, and m and n are arbitrary positive integers. Assume that $p_1 = p_2 = \frac{1}{2}$. Sketch the spectrum, and compare this result with the spectrum of the MSK signal.

3.17 Use the results in Section 3.4–4 to determine the power density spectrum of multitone FSK (MFSK) signals for which the signal waveforms are

$$s_n(t) = \sin \frac{2\pi n t}{T}, \qquad n = 1, 2, \ldots, M, \qquad 0 \le t \le T$$

Assume that the probabilities $p_n = 1/M$ for all n. Sketch the power spectral density.

3.18 A quadrature partial-response signal (QPRS) is generated by two separate partial-response signals of the type described in Problem 3.14 placed in phase quadrature. Hence, the QPRS is represented as

$$s(t) = \text{Re}\left[v(t)e^{j2\pi f_c t}\right]$$

where

$$v(t) = v_c(t) + j v_s(t) = \sum_n B_n g(t - nT) + j \sum_n C_n g(t - nT)$$

and $B_n = I_n + I_{n-1}$ and $C_n = J_n + J_{n-1}$. The sequences $\{B_n\}$ and $\{C_n\}$ are independent, and $I_n = \pm 1$, $J_n = \pm 1$ with equal probability.

1. Sketch the signal space diagram for the QPRS signal, and determine the probability of occurrence of each symbol.
2. Determine the autocorrelations and power spectral density of $v_c(t)$, $v_s(t)$, and $v(t)$.
3. Sketch the Markov chain model, and indicate the transition probabilities for the QPRS.

3.19 The information sequence $\{a_n\}_{n=-\infty}^{\infty}$ is a sequence of iid random variables, each taking values $+1$ and -1 with equal probability. This sequence is to be transmitted at baseband by a biphase coding scheme, described by

$$s(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT)$$

where $g(t)$ is shown in Figure P3.19.

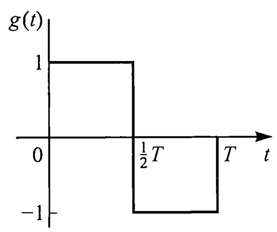


FIGURE P3.19

1. Find the power spectral density of $s(t)$.

2. Assume that it is desirable to have a zero in the power spectrum at $f = 1/T$. To this end, we use a precoding scheme by introducing $b_n = a_n + ka_{n-1}$, where k is some constant, and then transmit the $\{b_n\}$ sequence using the same $g(t)$. Is it possible to choose k to produce a frequency null at $f = 1/T$? If yes, what are the appropriate values and the resulting power spectrum?
3. Now assume we want to have zeros at all multiples of $f_0 = 1/4T$. Is it possible to have these zeros with an appropriate choice of k in the previous part? If not, then what kind of precoding do you suggest to achieve the desired result?

3.20 The two signal waveforms for binary FSK signal transmission with discontinuous phase are

$$s_0(t) = \sqrt{\frac{2E_b}{T_b}} \cos\left[2\pi\left(f_c - \frac{\Delta f}{2}\right)t + \theta_0\right], \quad 0 \le t < T$$

$$s_1(t) = \sqrt{\frac{2E_b}{T_b}} \cos\left[2\pi\left(f_c + \frac{\Delta f}{2}\right)t + \theta_1\right], \quad 0 \le t < T$$

where $\Delta f = 1/T \ll f_c$, and θ_0 and θ_1 are independent uniformly distributed random variables on the interval $(0, 2\pi)$. The signals $s_0(t)$ and $s_1(t)$ are equally probable.

1. Determine the power spectral density of the FSK signal.
2. Show that the power spectral density decays as $1/f^2$ for $f \gg f_c$.

3.21 The elements of the sequence $\{I_n\}_{n=-\infty}^{+\infty}$ are independent binary random variables taking values of ± 1 with equal probability. This data sequence is used to modulate the basic pulse $u(t)$ shown in Figure P3.21(a). The modulated signal is

$$X(t) = \sum_{n=-\infty}^{+\infty} I_n u(t - nT)$$

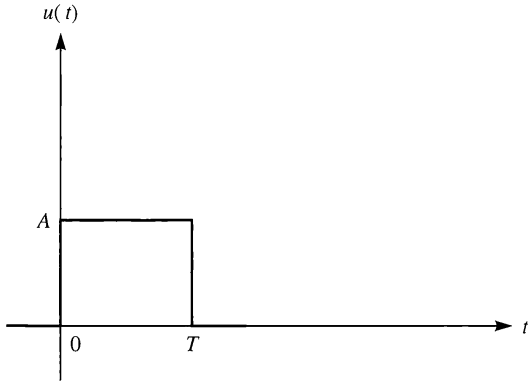


FIGURE P3.21(a)

1. Find the power spectral density of $X(t)$.
2. If $u_1(t)$, shown in Figure P3.21(b), were used instead of $u(t)$, how would the power spectrum in part 1 change?
3. In part 2, assume we want to have a null in the spectrum at $f = \frac{1}{3T}$. This is done by a precoding of the form $b_n = I_n + \alpha I_{n-1}$. Find the value of α that provides the desired null.

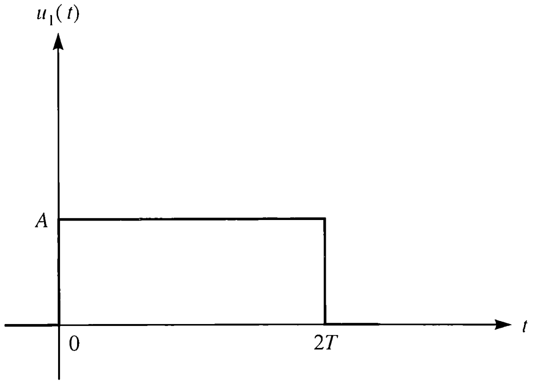


FIGURE P3.21(b)

4. Is it possible to employ a precoding of the form $b_n = I_n + \sum_{i=1}^{N} \alpha_i I_{n-i}$ for some finite N such that the final power spectrum will be identical to zero for $\frac{1}{3T} \le |f| \le \frac{1}{2T}$? If yes, how? If no, why? (*Hint:* Use properties of analytic functions.)

3.22 A digital signaling scheme is defined as

$$X(t) = \sum_{n=-\infty}^{\infty} [a_n u(t - nT)\cos(2\pi f_c t) - b_n u(t - nT)\sin(2\pi f_c t)]$$

where $u(t) = \Lambda(t/2T)$,

$$\Lambda(t) = \begin{cases} t+1 & -1 \le t \le 0 \\ -t+1 & 0 \le t \le 1 \\ 0 & \text{otherwise} \end{cases}$$

and each (a_n, b_n) pair is independent from the others and is equally likely to take any of the three values $(0, 1)$, $(\sqrt{3}/2, -1/2)$, and $(-\sqrt{3}/2, -1/2)$.

1. Determine the lowpass equivalent of the modulated signal. Determine the in-phase and quadrature components.
2. Determine the power spectral density of the lowpass equivalent signal; from this determine the power spectral density of the modulated signal.
3. By employing a precoding scheme of the form

$$\begin{cases} c_n = a_n + \alpha a_{n-1} \\ d_n = b_n + \alpha b_{n-1} \end{cases}$$

where α is in general a complex number, and transmitting the signal

$$Y(t) = \sum_{n=-\infty}^{\infty} [c_n u(t - nT)\cos(2\pi f_c t) - d_n u(t - nT)\sin(2\pi f_c t)]$$

we want to have a lowpass signal that has no dc component. Is it possible to achieve this goal by an appropriate choice of α? If yes, find this value.

3.23 A binary memoryless source generates the equiprobable outputs $\{a_k\}_{k=-\infty}^{\infty}$ which take values in $\{0, 1\}$. The source is modulated by mapping each sequence of length 3 of the source outputs into one of the eight possible $\{\alpha_i, \theta_i\}_{i=1}^{8}$ pairs and generating the modulated sequence

$$s(t) = \sum_{n=-\infty}^{\infty} \alpha_n g(t - nT)\cos(2\pi f_0 t + \theta_n)$$

where

$$g(t) = \begin{cases} 2t/T & 0 \leq t \leq T/2 \\ 2 - 2t/T & T/2 \leq t \leq T \\ 0 & \text{otherwise} \end{cases}$$

1. Find the power spectral density of $s(t)$ in terms of $\alpha^2 = \sum_{i=1}^{8} |\alpha_i|^2$ and $\beta = \sum_{i=1}^{8} \alpha_i e^{j\theta_i}$.
2. For the special case of $\alpha_{\text{odd}} = a$, $\alpha_{\text{even}} = b$, and $\theta_i = (i - 1)\pi/4$, determine the power spectral density of $s(t)$.
3. Show that for $a = b$, case 2 reduces to a standard 8-PSK signaling scheme, and determine the power spectrum in this case.
4. If a precoding of the form $b_n = a_n \oplus a_{n-1}$ (where $\oplus$ denotes the binary addition) were applied to the source outputs prior to modulation, how would the results in parts 1, 2, and 3 change?

3.24 An information source generates the ternary sequence $\{I_n\}_{n=-\infty}^{\infty}$. Each I_n can take one of the three possible values 2, 0, and -2 with probabilities 1/4, 1/2, and 1/4, respectively. The source outputs are assumed to be independent. The source outputs are used to generate the lowpass signal

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g(t - nT)$$

1. Determine the power spectral density of the process $v(t)$, assuming $g(t)$ is the signal shown in Figure P3.24.
2. Determine the power spectral density of

$$w(t) = \sum_{n=-\infty}^{\infty} J_n g(t - nT)$$

where $J_n = I_{n-1} + I_n + I_{n+1}$.

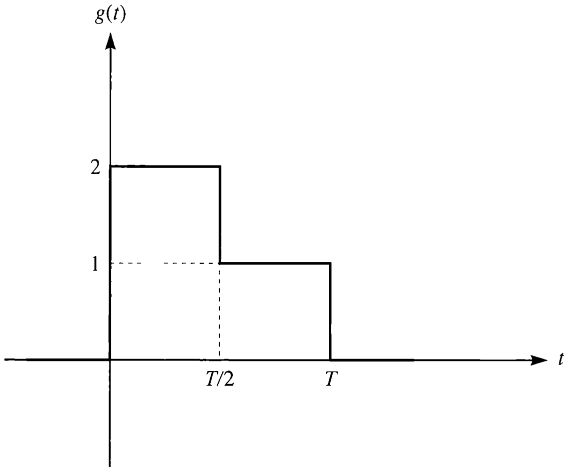


FIGURE P3.24

3.25 The information sequence $\{a_n\}$ is an iid sequence taking the values -1, 2, and 0 with probabilities 1/4, 1/4, and 1/2. This information sequence is used to generate the baseband signal

$$v(t) = \sum_{n=-\infty}^{\infty} a_n \operatorname{sinc}\left(\frac{t-nT}{T}\right)$$

1. Determine the power spectral density of $v(t)$.
2. Define the sequence $\{b_n\}$ as $b_n = a_n + a_{n-1} - a_{n-2}$ and generate the baseband signal

$$u(t) = \sum_{n=-\infty}^{\infty} b_n \operatorname{sinc}\left(\frac{t-nT}{T}\right)$$

   Determine the power spectral density of $u(t)$. What are the possible values for the b_n sequence?
3. Now let us assume $w(t)$ is defined as

$$w(t) = \sum_{n=-\infty}^{\infty} c_n \operatorname{sinc}\left(\frac{t-nT}{T}\right)$$

   where $c_n = a_n + ja_{n-1}$. Determine the power spectral density of $w(t)$.

(*Hint:* You can use the relation $\sum_{m=-\infty}^{\infty} e^{-j2\pi fmT} = \frac{1}{T}\sum_{m=-\infty}^{\infty} \delta(f - m/T)$.)

3.26 Let $\{a_n\}_{n=-\infty}^{\infty}$ denote an information sequence of independent random variables, taking values of ±1 with equal probability. A QPSK signal is generated by modulating a rectangular pulse shape of duration $2T$ by even and odd indexed a_n's to obtain the in-phase and quadrature components of the modulated signal. In other words, we have

$$g_{2T}(t) = \begin{cases} 1 & 0 \le t < 2T \\ 0 & \text{otherwise} \end{cases}$$

and we generate the in-phase and quadrature components according to

$$x_i(t) = \sum_{n=-\infty}^{\infty} a_{2n} g_{2T}(t - 2nT)$$

$$x_q(t) = \sum_{n=-\infty}^{\infty} a_{2n+1} g_{2T}(t - 2nT)$$

Then $x_l(t) = x_i(t) + jx_q(t)$ and $x(t) = \operatorname{Re}\left[x_l(t)e^{j2\pi f_0 t}\right]$.

1. Determine the power spectral density of $x_l(t)$.
2. Now let $x_q(t) = \sum_{n=-\infty}^{\infty} a_{2n+1} g_{2T}[t - (2n+1)T]$; in other words, let the quadrature component stagger the in-phase component by T. This results in an OQPSK system. Determine the power spectral density of $x_l(t)$ in this case. How does this compare with the result of part 1?
3. If in part 2 instead of $g_{2T}(t)$ we employ the following sinusoidal signal

$$g_1(t) = \begin{cases} \sin\left(\frac{\pi t}{2T}\right) & 0 \le t < 2T \\ 0 & \text{otherwise} \end{cases}$$

the resulting modulated signal will be an MSK signal. Determine the power spectral density of $x_l(t)$ in this case.

4. Show that in the case of MSK signaling, although the basic pulse $g_1(t)$ does not have a constant amplitude, the overall signal has a constant envelope.

3.27 $\{a_n\}_{n=-\infty}^{\infty}$ is a sequence of iid random variables each taking 0 or 1 with equal probability.

1. The sequence b_n is defined as $b_n = a_{n-1} \oplus a_n$ where $\oplus$ denotes binary addition (EXCLUSIVE-OR). Determine the autocorrelation function for the sequence b_n and the power spectral density of the PAM signal

$$v(t) = \sum_{n=-\infty}^{\infty} b_n g(t - nT)$$

where

$$g(t) = \begin{cases} 1 & 0 \le t < T \\ 0 & \text{otherwise} \end{cases}$$

2. Compare the result in part 1 with the result when $b_n = a_{n-1} + a_n$.

3.28 Consider the signal constellation shown in Figure P3.28.

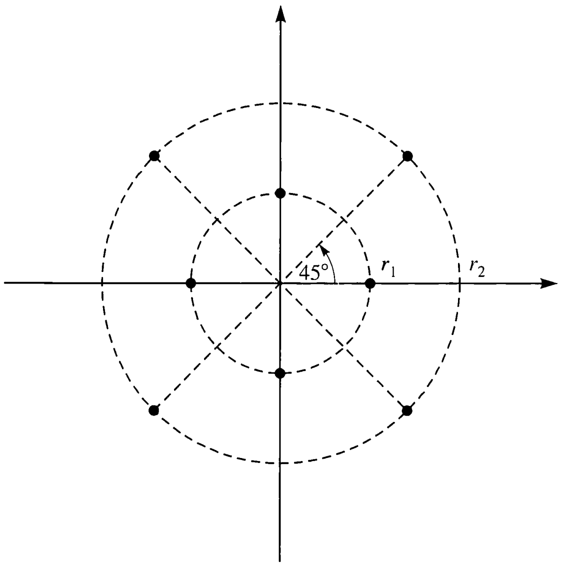


FIGURE P3.28

The lowpass equivalent of the transmitted signal is represented as

$$s_l(t) = \sum_{n=-\infty}^{\infty} a_n g(t - nT)$$

where $g(t)$ is a rectangular pulse defined as

$$g(t) = \begin{cases} 1 & 0 \le t < T \\ 0 & \text{otherwise} \end{cases}$$

and the a_n's are independent and identically distributed (iid) random variables that can assume the points in the constellation with equal probability.

1. Determine the power spectral density of the signal $s_l(t)$.
2. Determine the power spectral density of the transmitted signal $s(t)$, assuming that the carrier frequency is f_0 (assuming $f_0 \gg \frac{1}{T}$).
3. Determine and plot the power spectral density of $s_l(t)$ for the case when $r_1 = r_2$ (plot the PSD as a function of fT).

3.29 Determine the autocorrelation functions for the MSK and offset QPSK modulated signals based on the assumption that the information sequences for each of the two signals are uncorrelated and zero-mean.

3.30 Sketch the phase tree, the state trellis, and the state diagram for partial-response CPM with $h = \frac{1}{2}$ and

$$g(t) = \begin{cases} 1/4T & 0 \le t \le 2T \\ 0 & \text{otherwise} \end{cases}$$

3.31 Determine the number of terminal phase states in the state trellis diagram for

1. A full-response binary CPFSK with $h = \frac{2}{3}$ or $\frac{3}{4}$.
2. A partial-response $L = 3$ binary CPFSK with $h = \frac{2}{3}$ or $\frac{3}{4}$.

3.32 In the linear representation of CPM, show that the time durations of the 2^{L-1} pulses $\{c_k(t)\}$ are as follows:

$$\begin{aligned} c_0(t) &= 0, & t &< 0 \text{ and } t > (L+1)T \\ c_1(t) &= 0, & t &< 0 \text{ and } t > (L-1)T \\ c_2(t) = c_3(t) &= 0, & t &< 0 \text{ and } t > (L-2)T \\ c_4(t) = c_5(t) = c_6(t) = c_7(t) &= 0, & t &< 0 \text{ and } t > (L-3)T \\ &\vdots \\ c_{2^{L-2}}(t) = \cdots = c_{2^{L-1}}(t) &= 0, & t &< 0 \text{ and } t > T \end{aligned}$$

3.33 Use the result in Equation 3.4–31 to derive the expression for the power density spectrum of memoryless linear modulation given by Equation 3.4–16 under the condition that

$$s_k(t) = I_k s(t), \quad k = 1, 2, \ldots, K$$

where I_k is one of the K possible transmitted symbols that occur with equal probability.

3.34 Show that a sufficient condition for the absence of the line spectrum component in Equation 3.4–31 is

$$\sum_{i=1}^{K} p_i s_i(t) = 0$$

Is this condition necessary? Justify your answer.

4

Optimum Receivers for AWGN Channels

In Chapter 3, we described various types of modulation methods that may be used to transmit digital information through a communication channel. As we have observed, the modulator at the transmitter performs the function of mapping the information sequence into signal waveforms. These waveforms are transmitted over the channel, and a corrupted version of them is received at the receiver.

In Chapter 1 we have seen that communication channels can suffer from a variety of impairments that contribute to errors. These impairments include noise, attenuation, distortion, fading, and interference. Characteristics of a communication channel determine which impairments apply to that particular channel and which are the determining factors in the performance of the channel. Noise is present in all communication channels and is the major impairment in many communication systems. In this chapter we study the effect of noise on the reliability of the modulation systems studied in Chapter 3. In particular, this chapter deals with the design and performance characteristics of optimum receivers for the various modulation methods when the channel corrupts the transmitted signal by the addition of white Gaussian noise.

4.1

WAVEFORM AND VECTOR CHANNEL MODELS

The additive white Gaussian noise (AWGN) channel model is a channel whose sole effect is addition of a white Gaussian noise process to the transmitted signal. This channel is mathematically described by the relation

$$r(t) = s_m(t) + n(t) \tag{4.1–1}$$

where $s_m(t)$ is the transmitted signal which, as we have seen in Chapter 3 is one of M possible signals; $n(t)$ is a sample waveform of a zero-mean white Gaussian noise process with power spectral density of $N_0/2$; and $r(t)$ is the received waveform. This channel model is shown in Figure 4.1–1.

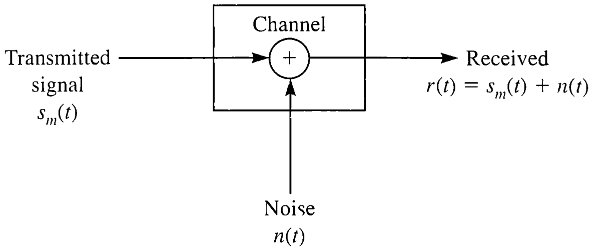


FIGURE 4.1–1
Model for received signal passed through an AWGN channel.

The receiver observes the received signal $r(t)$ and, based on this observation, makes the *optimal decision* about which message m, $1 \leq m \leq M$, was transmitted. By an optimal decision we mean a decision rule which results in minimum error probability, i.e., the decision rule that minimizes the probability of disagreement between the transmitted message m and the detected message $\hat{m}$ given by

$$P_e = \mathrm{P}[\hat{m} \neq m] \tag{4.1–2}$$

Although the AWGN channel model seems very limiting, its study is beneficial from two points of view. First, noise is the major type of corruption introduced by many channels. Therefore isolating it from other channel impairments and studying its effect results in better understanding of its effect on all communication systems. Second, the AWGN channel, although very simple, is a good model for studying deep space communication channels which were historically one of the first challenges encountered by communication engineers.

We have seen in Chapter 3 that by using an orthonormal basis $\{\phi_j(t),\ 1 \leq j \leq N\}$, each signal $s_m(t)$ can be represented by a vector $\boldsymbol{s}_m \in \mathbb{R}^N$. It was also shown in Example 2.8–1 that *any* orthonormal basis can be used for expansion of a zero-mean white Gaussian process, and the resulting coefficients of expansion will be iid zero-mean Gaussian random variables with variance $N_0/2$. Therefore, $\{\phi_j(t),\ 1 \leq j \leq N\}$, when extended appropriately, can be used for expansion of the noise process $n(t)$. This observation prompts us to view the waveform channel $r(t) = s_m(t) + n(t)$ in the vector form $\boldsymbol{r} = \boldsymbol{s}_m + \boldsymbol{n}$ where all vectors are N-dimensional and components of $\boldsymbol{n}$ are iid zero-mean Gaussian random variables with variance $N_0/2$. We will give a rigorous proof of this equivalence in Section 4.2. We continue our analysis with the study of the vector channel introduced above.

4.1–1 Optimal Detection for a General Vector Channel

The mathematical model for the AWGN vector channel is given by

$$\boldsymbol{r} = \boldsymbol{s}_m + \boldsymbol{n} \tag{4.1–3}$$

where all vectors are N-dimensional real vectors. The message m is chosen according to probabilities P_m from the set of possible messages $\{1, 2, \ldots, M\}$. The noise components n_j, $1 \leq j \leq N$, are iid, zero-mean, Gaussian random variables each distributed according to $\mathcal{N}(0, N_0/2)$. Therefore, the PDF of the noise vector $\boldsymbol{n}$ is given by

$$p(\boldsymbol{n}) = \left(\frac{1}{\sqrt{\pi N_0}}\right)^N e^{-\frac{\sum_{j=1}^{N} n_j^2}{2\sigma^2}} = \left(\frac{1}{\sqrt{\pi N_0}}\right)^N e^{-\frac{\|\boldsymbol{n}\|^2}{N_0}} \tag{4.1–4}$$

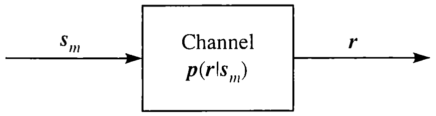


FIGURE 4.1–2
A general vector channel.

We, however, study a more general vector channel model in this section which is not limited to the AWGN channel model. This model will later be specialized to an AWGN channel model in Section 4.2. In our model, vectors $\boldsymbol{s}_m$ are selected from a set of possible signal vectors $\{\boldsymbol{s}_m,\ 1 \leq m \leq M\}$ according to *prior* or *a priori* probabilities P_m and transmitted over the channel. The received vector $\boldsymbol{r}$ depends statistically on the transmitted vector through the conditional probability density functions $p(\boldsymbol{r}|\boldsymbol{s}_m)$. The channel model is shown in Figure 4.1–2.

The receiver observes $\boldsymbol{r}$ and based on this observation decides which message was transmitted. Let us denote the decision function employed at the receiver by $g(\boldsymbol{r})$, which is a function from $\mathbb{R}^N$ into the set of messages $\{1, 2, \ldots, M\}$. Now if $g(\boldsymbol{r}) = \hat{m}$, i.e., the receiver decides that $\hat{m}$ was transmitted, then the probability that this decision is correct is the probability that $\hat{m}$ was in fact the transmitted message. In other words, the probability of a correct decision, given that $\boldsymbol{r}$ is received, is given by

$$\mathrm{P}[\text{correct decision}\,|\boldsymbol{r}] = \mathrm{P}[\hat{m}\text{ sent}\,|\boldsymbol{r}] \tag{4.1–5}$$

and therefore the probability of a correct decision is

$$\begin{aligned}\mathrm{P}[\text{correct decision}] &= \int \mathrm{P}[\text{correct decision}\,|\boldsymbol{r}]p(\boldsymbol{r})\,d\boldsymbol{r} \\ &= \int \mathrm{P}[\hat{m}\text{ sent}\,|\boldsymbol{r}]p(\boldsymbol{r})\,d\boldsymbol{r}\end{aligned} \tag{4.1–6}$$

Our goal is to design an optimal detector that minimizes the error probability or, equivalently, maximizes $\mathrm{P}[\text{correct decision}]$. Since $p(\boldsymbol{r})$ is nonnegative for all $\boldsymbol{r}$, the right-hand side of Equation 4.1–6 is maximized if for each $\boldsymbol{r}$ the quantity $\mathrm{P}[\hat{m}\,|\boldsymbol{r}]$ is maximized. This means that the optimal detection rule is the one that upon observing $\boldsymbol{r}$ decides in favor of the message m that maximizes $\mathrm{P}[m\,|\boldsymbol{r}]$. In other words,

$$\hat{m} = g_{\text{opt}}(\boldsymbol{r}) = \underset{1\leq m\leq M}{\arg\max}\, \mathrm{P}[m\,|\boldsymbol{r}] \tag{4.1–7}$$

The optimal detection scheme described in Equation 4.1–7 simply looks among all $\mathrm{P}[m\,|\boldsymbol{r}]$ for $1 \leq m \leq M$ and selects the m that maximizes $\mathrm{P}[m\,|\boldsymbol{r}]$. The detector then declares this maximizing m as its best decision. Note that since transmitting message m is equivalent to transmitting $\boldsymbol{s}_m$, the optimal decision rule can be written as

$$\hat{m} = g_{\text{opt}}(\boldsymbol{r}) = \underset{1\leq m\leq M}{\arg\max}\, \mathrm{P}[\boldsymbol{s}_m\,|\boldsymbol{r}] \tag{4.1–8}$$

MAP and ML Receivers

The optimal decision rule given by Equations 4.1–7 and 4.1–8 is known as the *maximum a posteriori probability* rule, or MAP rule. Note that the MAP receiver can be simplified to

$$\hat{m} = \underset{1 \le m \le M}{\arg\max} \frac{P_m p(\boldsymbol{r}|\boldsymbol{s}_m)}{p(\boldsymbol{r})} \tag{4.1–9}$$

and since $p(\boldsymbol{r})$ is independent of m and for all m remains the same, this is equivalent to

$$\hat{m} = \underset{1 \le m \le M}{\arg\max}\, P_m p(\boldsymbol{r}|\boldsymbol{s}_m) \tag{4.1–10}$$

Equation 4.1–10 is easier to use than Equation 4.1–7 since it is given in terms of the prior probabilities P_m and the probabilistic description of the channel $p(\boldsymbol{r}|\boldsymbol{s}_m)$, both directly known.

In the case where the messages are equiprobable a priori, i.e., when $P_m = \frac{1}{M}$ for all $1 \le m \le M$, the optimal detection rule reduces to

$$\hat{m} = \underset{1 \le m \le M}{\arg\max}\, p(\boldsymbol{r}|\boldsymbol{s}_m) \tag{4.1–11}$$

The term $p(\boldsymbol{r}|\boldsymbol{s}_m)$ is called the *likelihood* of message m, and the receiver given by Equation 4.1–11 is called the *maximum-likelihood receiver*, or ML receiver. It is important to note that the ML detector is not an optimal detector unless the messages are equiprobable. The ML detector, however, is a very popular detector since in many cases having exact information about message probabilities is difficult.

The Decision Regions

Any detector—including MAP and ML detectors—partitions the output space $\mathbb{R}^N$ into M regions denoted by $D_1, D_2, \ldots, D_M$ such that if $\boldsymbol{r} \in D_m$, then $\hat{m} = g(\boldsymbol{r}) = m$, i.e., the detector makes a decision in favor of m. The region D_m, $1 \le m \le M$, is called the *decision region* for message m; and D_m is the set of all outputs of the channel that are mapped into message m by the detector. If a MAP detector is employed, then the D_m's constitute the optimal decision regions resulting in the minimum possible error probability. For a MAP detector we have

$$D_m = \left\{\boldsymbol{r} \in \mathbb{R}^N : \mathrm{P}\left[m\,|\boldsymbol{r}\right] > \mathrm{P}\left[m'\,|\boldsymbol{r}\right],\ \text{for all } 1 \le m' \le M \text{ and } m' \ne m\right\} \tag{4.1–12}$$

Note that if for some given $\boldsymbol{r}$ two or more messages achieve the maximum a posteriori probability, we can arbitrarily assign $\boldsymbol{r}$ to one of the corresponding decision regions.

The Error Probability

To determine the error probability of a detection scheme, we note that when $\boldsymbol{s}_m$ is transmitted, an error occurs when the received $\boldsymbol{r}$ is not in D_m. The symbol error probability of a receiver with decision regions $\{D_m,\ 1 \le m \le M\}$ is therefore given by

$$\begin{aligned} P_e &= \sum_{m=1}^{M} P_m \mathrm{P}\left[\boldsymbol{r} \notin D_m\,|\boldsymbol{s}_m \text{ sent}\right] \\ &= \sum_{m=1}^{M} P_m P_{e|m} \end{aligned} \tag{4.1–13}$$

where $P_{e|m}$ denotes the error probability when message m is transmitted and is given by

$$\begin{aligned} P_{e|m} &= \int_{D_m^c} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \\ &= \sum_{\substack{1\le m'\le M\\ m'\ne m}} \int_{D_{m'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \end{aligned} \tag{4.1–14}$$

Using Equation 4.1–14 in Equation 4.1–13 gives

$$P_e = \sum_{m=1}^{M} P_m \sum_{\substack{1\le m'\le M\\ m'\ne m}} \int_{D_{m'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \tag{4.1–15}$$

Equation 4.1–15 gives the probability that an error occurs in transmission of a symbol or a message and is called *symbol error probability* or *message error probability*. Another type of error probability is the *bit error probability*. This error probability is denoted by P_b and is the error probability in transmission of a single bit. Determining the bit error probability in general requires detailed knowledge of how different bit sequences are mapped to signal points. Therefore, in general finding the bit error probability is not easy unless the constellation exhibits certain symmetry properties to make the derivation of the bit error probability easy. We will see later in this chapter that orthogonal signaling exhibits the required symmetry for calculation of the bit error probability. In other cases we can bound the bit error probability by noting that a symbol error occurs when at least one bit is in error, and the event of a symbol error is the union of the events of the errors in the $k = \log_2 M$ bits representing that symbol. Therefore we can write

$$P_b \le P_e \le kP_b \tag{4.1–16}$$

or

$$\frac{P_e}{\log_2 M} \le P_b \le P_e \tag{4.1–17}$$

EXAMPLE 4.1–1. Consider two equiprobable message signals $\boldsymbol{s}_1 = (0, 0)$ and $\boldsymbol{s}_2 = (1, 1)$. The channel adds iid noise components n_1 and n_2 to the transmitted vector each with an exponential PDF of the form

$$p(n) = \begin{cases} e^{-n} & n \ge 0 \\ 0 & n < 0 \end{cases}$$

Since the messages are equiprobable, the MAP detector is equivalent to the ML detector, and the decision region D_1 is given by

$$D_1 = \left\{ \boldsymbol{r} \in \mathbb{R}^2 : p(\boldsymbol{r}|\boldsymbol{s}_1) > p(\boldsymbol{r}|\boldsymbol{s}_2) \right\}$$

Noting that $p(\boldsymbol{r}|\boldsymbol{s} = (s_1, s_2)) = p(\boldsymbol{n} = \boldsymbol{r} - \boldsymbol{s})$, we have

$$D_1 = \left\{ \boldsymbol{r} \in \mathbb{R}^2 : p_{\boldsymbol{n}}(r_1, r_2) > p_{\boldsymbol{n}}(r_1 - 1, r_2 - 1) \right\}$$

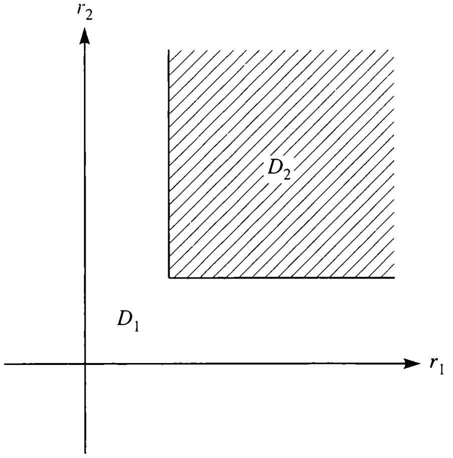


FIGURE 4.1–3
Decision regions D_1 and D_2.

where

$$p_{\boldsymbol{n}}(n_1, n_2) = \begin{cases} e^{-n_1-n_2} & n_1, n_2 > 0 \\ 0 & \text{otherwise} \end{cases}$$

From this relation we conclude that if either r_1 or r_2 is less than 1, then the point $\boldsymbol{r}$ belongs to D_1, and if both r_1 and r_2 are greater than 1, we have $e^{-r_1-r_2} < e^{-(r_1-1)-(r_2-1)}$ and $\boldsymbol{r}$ belongs to D_2.

Note that in this channel neither r_1 nor r_2 can be negative, because signal and noise are always nonnegative. Therefore,

$$D_2 = \left\{\boldsymbol{r} \in \mathbb{R}^2 : r_1 \geq 1, r_2 \geq 1\right\}$$

and

$$D_1 = \left\{\boldsymbol{r} \in \mathbb{R}^2 : r_1, r_2 \geq 0, \text{either } 0 \leq r_1 < 1 \text{ or } 0 \leq r_2 < 1\right\}$$

The decision regions are shown in Figure 4.1–3. For this channel, when $\boldsymbol{s}_2$ is transmitted, regardless of the value of noise components, $\boldsymbol{r}$ will always be in D_2 and no error will occur.

Errors will occur only when $\boldsymbol{s}_1 = (0, 0)$ is transmitted and the received vector $\boldsymbol{r}$ belongs to D_2, i.e., when both noise components exceed 1. Therefore, the error probability is given by

$$\begin{aligned} P_e &= \frac{1}{2}\,\mathrm{P}\left[\boldsymbol{r} \in D_2 \,|\, \boldsymbol{s}_1 = (0,0) \text{ sent}\right] \\ &= \frac{1}{2}\int_1^{\infty} e^{-n_1}\,dn_1 \int_1^{\infty} e^{-n_2}\,dn_2 \\ &= \frac{1}{2}e^{-2} \approx 0.0068 \end{aligned}$$

Sufficient Statistics

Let us assume that at the receiver we have access to a vector $\boldsymbol{r}$ that can be written in terms of two vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$, i.e., $\boldsymbol{r} = (\boldsymbol{r}_1, \boldsymbol{r}_2)$. We further assume that $\boldsymbol{s}_m$, $\boldsymbol{r}_1$, and $\boldsymbol{r}_2$ constitute a Markov chain in the given order, i.e.,

$$p(\boldsymbol{r}_1, \boldsymbol{r}_2|\boldsymbol{s}_m) = p(\boldsymbol{r}_1|\boldsymbol{s}_m)p(\boldsymbol{r}_2|\boldsymbol{r}_1) \qquad (4.1\text{–}18)$$

Under these assumptions $\boldsymbol{r}_2$ can be ignored in the detection of s_m, and the detection can be based only on $\boldsymbol{r}_1$. The reason is that by Equation 4.1–10

$$\begin{aligned}\hat{m} &= \underset{1\le m\le M}{\arg\max}\, P_m p(\boldsymbol{r}|\boldsymbol{s}_m)\\ &= \underset{1\le m\le M}{\arg\max}\, P_m p(\boldsymbol{r}_1, \boldsymbol{r}_2|\boldsymbol{s}_m)\\ &= \underset{1\le m\le M}{\arg\max}\, P_m p(\boldsymbol{r}_1|\boldsymbol{s}_m)p(\boldsymbol{r}_2|\boldsymbol{r}_1)\\ &= \underset{1\le m\le M}{\arg\max}\, P_m p(\boldsymbol{r}_1|\boldsymbol{s}_m)\end{aligned} \tag{4.1–19}$$

where in the last step we have ignored the positive factor $p(\boldsymbol{r}_2|\boldsymbol{r}_1)$ since it does not depend on m. This shows that the optimal detection can be based only on $\boldsymbol{r}_1$.

When the Markov chain relation among s_m, $\boldsymbol{r}_1$, and $\boldsymbol{r}_2$ as given in Equation 4.1–18 is satisfied, it is said that $\boldsymbol{r}_1$ is a *sufficient statistic* for detection of s_m. In such a case, when $\boldsymbol{r}_2$ can be ignored without sacrificing the optimality of the receiver, $\boldsymbol{r}_2$ is called *irrelevant data* or *irrelevant information.* Recognizing sufficient statistics helps to reduce the complexity of the detection process through ignoring a usually large amount of irrelevant data at the receiver.

EXAMPLE 4.1–2. Let us assume that in Example 4.1–1, in addition to $\boldsymbol{r}$, the receiver can observe n_1 as well. Therefore, we can assume that $\boldsymbol{r} = (\boldsymbol{r}_1, \boldsymbol{r}_2)$ is available at the receiver, where $\boldsymbol{r}_1 = (r_1, n_1)$ and $\boldsymbol{r}_2 = r_2$. To design the optimal detector, we notice that having access to both r_1 and n_1 uniquely determines s_{m1} at the receiver; and since $s_{11} = 0$ and $s_{21} = 1$, this uniquely determines the message m, thus making $\boldsymbol{r}_2 = r_2$ irrelevant. The optimal decision rule in this case becomes

$$\hat{m} = \begin{cases} 1 & \text{if } r_1 - n_1 = 0\\ 2 & \text{if } r_1 - n_1 = 1\end{cases} \tag{4.1–20}$$

and the resulting error probability is zero.

Preprocessing at the Receiver

Let us assume that the receiver applies an invertible operation $\boldsymbol{G}(\cdot)$ on the received vector $\boldsymbol{r}$. In other words instead of supplying $\boldsymbol{r}$ to the detector, the receiver passes $\boldsymbol{r}$ through $\boldsymbol{G}$ and supplies the detector with $\boldsymbol{\rho} = \boldsymbol{G}(\boldsymbol{r})$, as shown in Figure 4.1–4.

Since $\boldsymbol{G}$ is invertible and the detector has access to $\boldsymbol{\rho}$, it can apply $\boldsymbol{G}^{-1}$ to $\boldsymbol{\rho}$ to obtain $\boldsymbol{G}^{-1}(\boldsymbol{\rho}) = \boldsymbol{G}^{-1}(\boldsymbol{G}(\boldsymbol{r})) = \boldsymbol{r}$. The detector now has access to both $\boldsymbol{\rho}$ and $\boldsymbol{r}$; therefore the

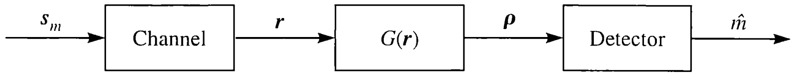


FIGURE 4.1–4
Preprocessing at the receiver.

optimal detection rule is

$$\begin{aligned}\hat{m} &= \arg\max_{1\le m\le M} P_m p(\boldsymbol{r}, \boldsymbol{\rho}|\boldsymbol{s}_m) \\ &= \arg\max_{1\le m\le M} P_m p(\boldsymbol{r}|\boldsymbol{s}_m) p(\boldsymbol{\rho}|\boldsymbol{r}) \\ &= \arg\max_{1\le m\le M} P_m p(\boldsymbol{r}|\boldsymbol{s}_m)\end{aligned} \tag{4.1–21}$$

where we have used the fact that $\boldsymbol{\rho}$ is a function of $\boldsymbol{r}$ and hence, when $\boldsymbol{r}$ is given, $\boldsymbol{\rho}$ does not depend on $\boldsymbol{s}_m$. From Equation 4.1–21 it is clear that the optimal detector based on the observation of $\boldsymbol{\rho}$ makes the same decision as the optimal detector based on the observation of $\boldsymbol{r}$. In other words, an invertible preprocessing of the received information does not change the optimality of the receiver.

EXAMPLE 4.1–3. Let us assume the received vector is of the form

$$\boldsymbol{r} = \boldsymbol{s}_m + \boldsymbol{n}$$

where $\boldsymbol{n}$ is a nonwhite (colored) noise. Let us further assume that there exists an invertible whitening operator denoted by matrix $\boldsymbol{W}$ such that $\boldsymbol{\nu} = \boldsymbol{W}\boldsymbol{n}$ is a white vector. Then we can consider

$$\boldsymbol{\rho} = \boldsymbol{W}\boldsymbol{r} = \boldsymbol{W}\boldsymbol{s}_m + \boldsymbol{\nu}$$

which is equivalent to a channel with white noise for detection without degrading the performance. The linear operation denoted by $\boldsymbol{W}$ is called a *whitening filter*.

4.2
WAVEFORM AND VECTOR AWGN CHANNELS

The waveform AWGN channel is described by the input-output relation

$$r(t) = s_m(t) + n(t) \tag{4.2–1}$$

where $s_m(t)$ is one of the possible M signals $\{s_1(t), s_2(t), \dots, s_M(t)\}$, each selected with prior probability P_m and $n(t)$ is a zero-mean white Gaussian process with power spectral density $\frac{N_0}{2}$. Let us assume that using the Gram-Schmidt procedure, we have derived an orthonormal basis $\{\phi_j(t),\ 1 \le j \le N\}$ for representation of the signals and, using this set, the vector representation of the signals is given by $\{\boldsymbol{s}_m,\ 1 \le m \le M\}$. The noise process cannot be completely expanded in terms of the basis $\{\phi_j(t)\}_{j=1}^N$. We decompose the noise process $n(t)$ into two components. One component, denoted by $n_1(t)$ is part of the noise process that can be expanded in terms of $\{\phi_j(t)\}_{j=1}^N$, i.e., the projection of the noise onto the space spanned by these basis functions; and the other part, denoted by $n_2(t)$, is the part that cannot be expressed in terms of this basis function. With this definition we have

$$n_1(t) = \sum_{j=1}^{N} n_j \phi_j(t), \qquad \text{where } n_j = \langle n(t), \phi_j(t)\rangle \tag{4.2–2}$$

and

$$n_2(t) = n(t) - n_1(t) \tag{4.2–3}$$

Noting that

$$s_m(t) = \sum_{j=1}^{N} s_{mj}\phi_j(t), \qquad \text{where } s_{mj} = \langle s_m(t), \phi_j(t)\rangle \tag{4.2–4}$$

and using Equations 4.2–2 and 4.2–3, we can write Equation 4.2–1 as

$$r(t) = \sum_{j=1}^{N} (s_{mj} + n_j)\phi_j(t) + n_2(t) \tag{4.2–5}$$

By defining

$$r_j = s_{mj} + n_j \tag{4.2–6}$$

where

$$r_j = \langle s_m(t), \phi_j(t)\rangle + \langle n(t), \phi_j(t)\rangle = \langle s_m(t) + n(t), \phi_j(t)\rangle = \langle r(t), \phi_j(t)\rangle \tag{4.2–7}$$

we have

$$r(t) = \sum_{j=1}^{N} r_j\phi_j(t) + n_2(t), \qquad \text{where } r_j = \langle r(t), \phi_j(t)\rangle \tag{4.2–8}$$

From Example 2.8–1 we know that n_j's are iid zero-mean Gaussian random variables each with variance $\frac{N_0}{2}$. This result can also be directly shown, by noting that the n_j's defined by

$$n_j = \int_{-\infty}^{\infty} n(t)\phi_j(t)\,dt \tag{4.2–9}$$

are linear combinations of the Gaussian random process $n(t)$, and therefore they are Gaussian. Their mean is given by

$$\begin{aligned} \mathrm{E}[n_j] &= \mathrm{E}\left[\int_{-\infty}^{\infty} n(t)\phi_j(t)\,dt\right] \\ &= \int_{-\infty}^{\infty} \mathrm{E}[n(t)]\,\phi_j(t)\,dt \\ &= 0 \end{aligned} \tag{4.2–10}$$

where the last equality holds since $n(t)$ is zero-mean, i.e., $\mathrm{E}[n(t)] = 0$.

We can also find the covariance of n_i and n_j as

$$\begin{aligned}
\operatorname{COV}[n_i n_j] &= \mathrm{E}[n_i n_j] - \mathrm{E}[n_i]\mathrm{E}[n_j] \\
&= \mathrm{E}\left[\int_{-\infty}^{\infty} n(t)\phi_i(t)\,dt \int_{-\infty}^{\infty} n(s)\phi_j(s)\,ds\right] \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{E}[n(t)n(s)]\,\phi_i(t)\phi_j(s)\,dt\,ds \\
&= \frac{N_0}{2}\int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty}\delta(t-s)\phi_i(t)\,dt\right]\phi_j(s)\,ds \\
&= \frac{N_0}{2}\int_{-\infty}^{\infty}\phi_i(s)\phi_j(s)\,ds \\
&= \begin{cases} \frac{N_0}{2} & i = j \\ 0 & i \neq j \end{cases}
\end{aligned} \tag{4.2–11}$$

where we have used the facts that n_i and n_j are zero-mean, and since $n(t)$ is white, its autocorrelation function is $\frac{N_0}{2}\delta(\tau)$. In the last step we applied the orthonormality of $\{\phi_j(t)\}$. Equation 4.2–11 shows that for $i \neq j$, n_i and n_j are uncorrelated and since they are Gaussian, they are independent as well. It also shows that each n_j has a variance equal to $\frac{N_0}{2}$.

Now we study the properties of $n_2(t)$. We first observe that since the n_j's are jointly Gaussian random variables, the process $n_1(t)$ is a Gaussian process and thus $n_2(t) = n(t) - n_1(t)$, which is a linear combination of two jointly Gaussian processes, is itself a Gaussian process. At any given t we have

$$\begin{aligned}
\operatorname{COV}[n_j n_2(t)] &= \mathrm{E}[n_j n_2(t)] \\
&= \mathrm{E}[n_j n(t)] - \mathrm{E}[n_j n_1(t)] \\
&= \mathrm{E}\left[n(t)\int_{-\infty}^{\infty} n(s)\phi_j(s)\,ds\right] - \mathrm{E}\left[n_j\sum_{i=1}^{N} n_i\phi_i(t)\right] \\
&= \frac{N_0}{2}\int_{-\infty}^{\infty}\delta(t-s)\phi_j(s)\,ds - \frac{N_0}{2}\phi_j(t) \\
&= \frac{N_0}{2}\phi_j(t) - \frac{N_0}{2}\phi_j(t) \\
&= 0
\end{aligned} \tag{4.2–12}$$

where we have used the fact that $\mathrm{E}[n_j n_i] = 0$, except when $i = j$, in which case $\mathrm{E}[n_j n_j] = N_0/2$.

Equation 4.2–12 shows that $n_2(t)$ is uncorrelated with all n_j's, and since they are jointly Gaussian, $n_2(t)$ is independent of all n_j's, and therefore it is independent of $n_1(t)$.

Since $n_2(t)$ is independent of $s_m(t)$ and $n_1(t)$, we conclude that in Equation 4.2–8, the two components of $r(t)$, namely, $\sum_j r_j\phi_j(t)$ and $n_2(t)$, are independent. Since the first component is the only component that carries the transmitted signal, and the second component is independent of the first component, the second component cannot provide any information about the transmitted signal and therefore has no effect in the detection process and can be ignored without sacrificing the optimality of the detector. In other words $n_2(t)$ is irrelevant information for optimal detection.

From the above discussion it is clear that for the design of the optimal detector, the AWGN waveform channel of the form

$$r(t) = s_m(t) + n(t), \qquad 1 \leq m \leq M \tag{4.2–13}$$

is equivalent to the N-dimensional vector channel

$$\boldsymbol{r} = \boldsymbol{s}_m + \boldsymbol{n}, \qquad 1 \leq m \leq M \tag{4.2–14}$$

4.2–1 Optimal Detection for the Vector AWGN Channel

The additive AWGN vector channel is the vector equivalent channel to the waveform AWGN channel and is described by Equation 4.2–14 in which the components of the noise vector are iid zero-mean Gaussian random variables with variance $\frac{N_0}{2}$. The joint PDF of the noise vector is given by Equation 4.1–4. The MAP detector for this channel is given by

$$\begin{aligned}
\hat{m} &= \underset{1\leq m\leq M}{\arg\max}\, [P_m p(\boldsymbol{r}|\boldsymbol{s}_m)] \\
&= \underset{1\leq m\leq M}{\arg\max}\, P_m \left[p_{\boldsymbol{n}}(\boldsymbol{r} - \boldsymbol{s}_m)\right] \\
&= \underset{1\leq m\leq M}{\arg\max} \left[P_m \left(\frac{1}{\sqrt{\pi N_0}}\right)^N e^{-\frac{\|\boldsymbol{r}-\boldsymbol{s}_m\|^2}{N_0}}\right] \\
&\overset{(a)}{=} \underset{1\leq m\leq M}{\arg\max} \left[P_m\, e^{-\frac{\|\boldsymbol{r}-\boldsymbol{s}_m\|^2}{N_0}}\right] \\
&\overset{(b)}{=} \underset{1\leq m\leq M}{\arg\max} \left[\ln P_m - \frac{\|\boldsymbol{r} - \boldsymbol{s}_m\|^2}{N_0}\right] \\
&\overset{(c)}{=} \underset{1\leq m\leq M}{\arg\max} \left[\frac{N_0}{2}\ln P_m - \frac{1}{2}\|\boldsymbol{r} - \boldsymbol{s}_m\|^2\right] \\
&= \underset{1\leq m\leq M}{\arg\max} \left[\frac{N_0}{2}\ln P_m - \frac{1}{2}\left(\|\boldsymbol{r}\|^2 + \|\boldsymbol{s}_m\|^2 - 2\boldsymbol{r}\cdot\boldsymbol{s}_m\right)\right] \\
&\overset{(d)}{=} \underset{1\leq m\leq M}{\arg\max} \left[\frac{N_0}{2}\ln P_m - \frac{1}{2}\mathcal{E}_m + \boldsymbol{r}\cdot\boldsymbol{s}_m\right] \\
&\overset{(e)}{=} \underset{1\leq m\leq M}{\arg\max}\, [\eta_m + \boldsymbol{r}\cdot\boldsymbol{s}_m]
\end{aligned} \tag{4.2–15}$$

where we have used the following steps in simplifying the expression:

(a): $\left(\frac{1}{\sqrt{\pi N_0}}\right)^N$ is a positive constant and can be dropped.

(b): $\ln(\cdot)$ is an increasing function.

(c): $\frac{N_0}{2}$ is positive and multiplying by a positive number does not affect the result of arg max.
(d): $\|\boldsymbol{r}\|^2$ was dropped since it does not depend on m and $\|\boldsymbol{s}_m\|^2 = \mathcal{E}_m$.
(e): We have defined

$$\eta_m = \frac{N_0}{2} \ln P_m - \frac{1}{2}\mathcal{E}_m \tag{4.2–16}$$

as the *bias term.*

From Equation 4.2–15, it is clear that the optimal (MAP) decision rule for an AWGN vector channel is given by

$$\begin{aligned} \hat{m} &= \underset{1 \le m \le M}{\arg\max}\, [\eta_m + \boldsymbol{r} \cdot \boldsymbol{s}_m] \\ \eta_m &= \frac{N_0}{2} \ln P_m - \frac{1}{2}\mathcal{E}_m \end{aligned} \tag{4.2–17}$$

In the special case where the signals are equiprobable, i.e., $P_m = 1/M$ for all m, this relation becomes somewhat simpler. In this case Equation 4.2–15 at step (c) can be written as

$$\begin{aligned} \hat{m} &= \underset{1 \le m \le M}{\arg\max} \left[\frac{N_0}{2} \ln P_m - \frac{1}{2}\|\boldsymbol{r} - \boldsymbol{s}_m\|^2\right] \\ &= \underset{1 \le m \le M}{\arg\max} \left[-\|\boldsymbol{r} - \boldsymbol{s}_m\|^2\right] \\ &= \underset{1 \le m \le M}{\arg\min}\, \|\boldsymbol{r} - \boldsymbol{s}_m\| \end{aligned} \tag{4.2–18}$$

where we have used the fact that maximizing $-\|\boldsymbol{r} - \boldsymbol{s}_m\|^2$ is equivalent to minimizing its negative, i.e., $\|\boldsymbol{r} - \boldsymbol{s}_m\|^2$, which is equivalent to minimizing its square root $\|\boldsymbol{r} - \boldsymbol{s}_m\|$.

A geometric interpretation of Equation 4.2–18 is particularly convenient. The receiver receives $\boldsymbol{r}$ and looks among all $\boldsymbol{s}_m$ to find the one that is closest to $\boldsymbol{r}$ using standard Euclidean distance. Such a detector is called a *nearest-neighbor,* or *minimum-distance,* detector. Also note that in this case, since the signals are equiprobable, the MAP and the ML detector coincide, and both are equivalent to the minimum-distance detector. In this case the boundaries of decisions D_m and $D_{m'}$ are the set of points that are equidistant from $\boldsymbol{s}_m$ and $\boldsymbol{s}_{m'}$, which is the perpendicular bisector of the line connecting these two signal points. This boundary in general is a hyperplane. For the case of $N = 2$ the boundary is a line, and for $N = 3$ it is a plane. These hyperplanes completely determine the decision regions. An example of a two-dimensional constellation ($N = 2$) with four signal points ($M = 4$) is shown in Figure 4.2–1. The solid lines denote the boundaries of the decision regions which are the perpendicular bisectors of the dashed lines connecting the signal points.

When the signals are both equiprobable and have equal energy, the bias terms defined as $\eta_m = \frac{N_0}{2} \ln P_m - \frac{1}{2}\mathcal{E}_m$ are independent of m and can be dropped from Equation 4.2–17. The optimal detection rule in this case reduces to

$$\hat{m} = \underset{1 \le m \le M}{\arg\max}\ \boldsymbol{r} \cdot \boldsymbol{s}_m \tag{4.2–19}$$

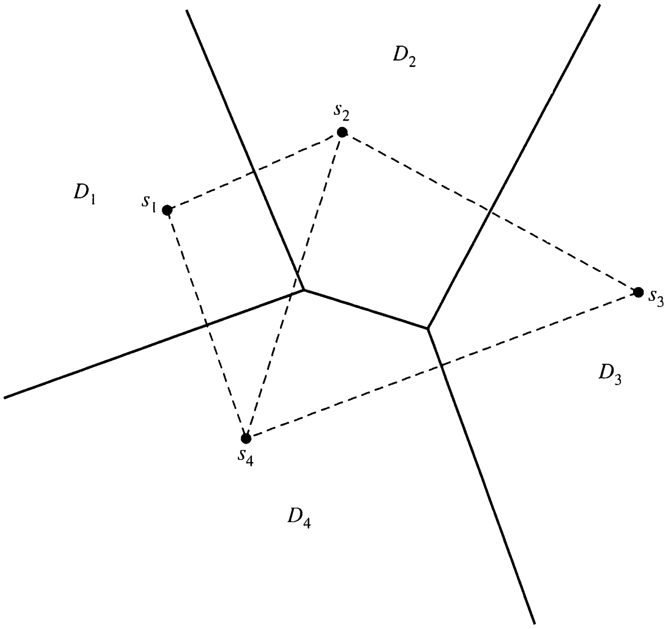


FIGURE 4.2–1
The decision regions for equiprobable signaling.

In general, the decision region D_m is given as

$$D_m = \left\{ \boldsymbol{r} \in \mathbb{R}^N : \boldsymbol{r} \cdot \boldsymbol{s}_m + \eta_m > \boldsymbol{r} \cdot \boldsymbol{s}_{m'} + \eta_{m'}, \quad \text{for all } 1 \le m' \le M \text{ and } m' \ne m \right\} \tag{4.2–20}$$

Note that each decision region is described in terms of at most $M - 1$ inequalities. In some cases some of these inequalities are dominated by the others and are redundant. Also note that each boundary is of the general form of

$$\boldsymbol{r} \cdot (\boldsymbol{s}_m - \boldsymbol{s}_{m'}) > \eta_{m'} - \eta_m \tag{4.2–21}$$

which is the equation of a hyperplane. Therefore the boundaries of the decision regions in general are hyperplanes.

From Equation 2.2–47, we know that

$$\boldsymbol{r} \cdot \boldsymbol{s}_m = \int_{-\infty}^{\infty} r(t) s_m(t)\, dt \tag{4.2–22}$$

and

$$\mathcal{E}_m = \|s\|^2 = \int_{-\infty}^{\infty} s_m^2(t)\, dt \tag{4.2–23}$$

Therefore, the optimal MAP detection rule in an AWGN channel can be written in the form

$$\hat{m} = \underset{1 \le m \le M}{\arg\max} \left[\frac{N_0}{2} \ln P_m + \int_{-\infty}^{\infty} r(t) s_m(t)\, dt - \frac{1}{2} \int_{-\infty}^{\infty} s_m^2(t)\, dt \right] \tag{4.2–24}$$

and the ML detector has the following form:

$$\hat{m} = \arg\max_{1\le m\le M} \left[\int_{-\infty}^{\infty} r(t)s_m(t)\,dt - \frac{1}{2}\int_{-\infty}^{\infty} s_m^2(t)\,dt \right] \tag{4.2–25}$$

At this point it is convenient to introduce three metrics that we will use frequently in the future. We define the *distance metric* as

$$\begin{aligned} D(\boldsymbol{r}, \boldsymbol{s}_m) &= \|\boldsymbol{r} - \boldsymbol{s}_m\|^2 \\ &= \int_{-\infty}^{\infty} (r(t) - s_m(t))^2\,dt \end{aligned} \tag{4.2–26}$$

denoting the square of the Euclidean distance between $\boldsymbol{r}$ and $\boldsymbol{s}_m$. The *modified distance metric* is defined as

$$D'(\boldsymbol{r}, \boldsymbol{s}_m) = -2\boldsymbol{r}\cdot\boldsymbol{s}_m + \|\boldsymbol{s}_m\|^2 \tag{4.2–27}$$

and is equal to the distance metric when the term $\|\boldsymbol{r}\|^2$, which does not depend on m, is removed. The *correlation metric* is defined as the negative of the modified distance metric and is given by

$$\begin{aligned} C(\boldsymbol{r}, \boldsymbol{s}_m) &= 2\boldsymbol{r}\cdot\boldsymbol{s}_m - \|\boldsymbol{s}_m\|^2 \\ &= 2\int_{-\infty}^{\infty} r(t)s_m(t)\,dt - \int_{-\infty}^{\infty} s_m^2(t)\,dt \end{aligned} \tag{4.2–28}$$

It is important to note that using the term *metric* is just for convenience. In general, none of these quantities is a metric in a mathematical sense. With these definitions the optimal detection rule (MAP rule) in general can be written as

$$\begin{aligned} \hat{m} &= \arg\max_{1\le m\le M} [N_0 \ln P_m - D(\boldsymbol{r}, \boldsymbol{s}_m)] \\ &= \arg\max_{1\le m\le M} [N_0 \ln P_m + C(\boldsymbol{r}, \boldsymbol{s}_m)] \end{aligned} \tag{4.2–29}$$

and the ML detection rule becomes

$$\hat{m} = \arg\max_{1\le m\le M} C(\boldsymbol{r}, \boldsymbol{s}_m) \tag{4.2–30}$$

Optimal Detection for Binary Antipodal Signaling

In a binary antipodal signaling scheme $s_1(t) = s(t)$ and $s_2(t) = -s(t)$. The probabilities of messages 1 and 2 are p and $1-p$, respectively. This is obviously a case with $N = 1$, and the vector representations of the two signals are just scalars with $s_1 = \sqrt{\mathcal{E}_s}$ and $s_2 = -\sqrt{\mathcal{E}_s}$, where $\mathcal{E}_s$ is energy in each signal and is equal to $\mathcal{E}_b$. Following Equation 4.2–20, the decision region D_1 is given as

$$\begin{aligned} D_1 &= \left\{ r : r\sqrt{\mathcal{E}_b} + \frac{N_0}{2}\ln p - \frac{1}{2}\mathcal{E}_b > -r\sqrt{\mathcal{E}_b} + \frac{N_0}{2}\ln(1-p) - \frac{1}{2}\mathcal{E}_b \right\} \\ &= \left\{ r : r > \frac{N_0}{4\sqrt{\mathcal{E}_b}} \ln\frac{1-p}{p} \right\} \\ &= \{ r : r > r_{\text{th}} \} \end{aligned} \tag{4.2–31}$$

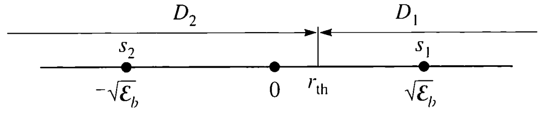


FIGURE 4.2–2
The decision regions for antipodal signaling.

where the threshold r_{th} is defined as

$$r_{\text{th}} = \frac{N_0}{4\sqrt{\mathcal{E}_b}} \ln \frac{1-p}{p} \tag{4.2–32}$$

The constellation and the decision regions are shown in Figure 4.2–2.

Note that as $p \to 0$, we have $r_{\text{th}} \to \infty$ and the entire real line becomes D_2; and when $p \to 1$, the entire line becomes D_1, as expected. Also note that when $p = \frac{1}{2}$, i.e., when the messages are equiprobable, $r_{\text{th}} = 0$ and the decision rule reduces to a minimum-distance rule. To derive the error probability for this system, we use Equation 4.1–15. This yields

$$\begin{aligned}
P_e &= \sum_{m=1}^{2} P_m \sum_{\substack{1 \le m' \le 2 \\ m' \ne m}} \int_{D_{m'}} p(\boldsymbol{r} \,|\boldsymbol{s}_m)\, d\boldsymbol{r} \\
&= p \int_{D_2} p\left(r \,\middle|\, s = \sqrt{\mathcal{E}_b}\right) dr + (1-p) \int_{D_1} p\left(r \,\middle|\, s = -\sqrt{\mathcal{E}_b}\right) dr \\
&= p \int_{-\infty}^{r_{\text{th}}} p\left(r \,\middle|\, s = \sqrt{\mathcal{E}_b}\right) dr + (1-p) \int_{r_{\text{th}}}^{\infty} p\left(r \,\middle|\, s = -\sqrt{\mathcal{E}_b}\right) dr \\
&= p\,\mathrm{P}\left[\mathcal{N}\left(\sqrt{\mathcal{E}_b}, \frac{N_0}{2}\right) < r_{\text{th}}\right] + (1-p)\,\mathrm{P}\left[\mathcal{N}\left(-\sqrt{\mathcal{E}_b}, \frac{N_0}{2}\right) > r_{\text{th}}\right] \\
&= pQ\left(\frac{\sqrt{\mathcal{E}_b} - r_{\text{th}}}{\sqrt{\frac{N_0}{2}}}\right) + (1-p)Q\left(\frac{r_{\text{th}} + \sqrt{\mathcal{E}_b}}{\sqrt{\frac{N_0}{2}}}\right)
\end{aligned} \tag{4.2–33}$$

where in the last step we have used Equation 2.3–12. In the special case where $p = \frac{1}{2}$, we have $r_{\text{th}} = 0$ and the error probability simplifies to

$$P_e = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{4.2–34}$$

Also note that since the system is binary, the error probability for each message is equal to the bit error probability, i.e., $P_b = P_e$.

Error Probability for Equiprobable Binary Signaling Schemes

In this case the transmitter transmits one of the two equiprobable signals $s_1(t)$ and $s_2(t)$ over the AWGN channel. Since the signals are equiprobable, the two decision regions are separated by the perpendicular bisector of the line connecting $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$. By symmetry, error probabilities when $\boldsymbol{s}_1$ or $\boldsymbol{s}_2$ is transmitted are equal, therefore $P_b = \mathrm{P}\,[\text{error}\,|\boldsymbol{s}_1 \text{ sent}]$. The decision regions and the perpendicular bisector of the line connecting $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are shown in Figure 4.2–3.

Since we are assuming that $\boldsymbol{s}_1$ is sent, an error occurs if $\boldsymbol{r}$ is in D_2, which means the distance between the projection of $\boldsymbol{r} - \boldsymbol{s}_1$ on $\boldsymbol{s}_2 - \boldsymbol{s}_1$, i.e., point A, from $\boldsymbol{s}_1$ is larger than

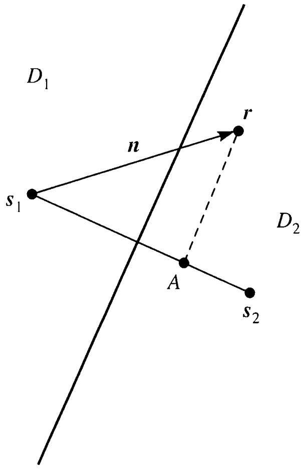


FIGURE 4.2–3
Decision regions for binary equiprobable signals.

$\frac{d_{12}}{2}$, where $d_{12} = \|\boldsymbol{s}_2 - \boldsymbol{s}_1\|$. Note that since $\boldsymbol{s}_1$ is sent, $\boldsymbol{n} = \boldsymbol{r} - \boldsymbol{s}_1$, and the projection of $\boldsymbol{r} - \boldsymbol{s}_1$ on $\boldsymbol{s}_2 - \boldsymbol{s}_1$ becomes equal to $\frac{\boldsymbol{n} \cdot (\boldsymbol{s}_2 - \boldsymbol{s}_1)}{d_{12}}$. Therefore, the error probability is given by

$$P_b = \mathrm{P}\left[\frac{\boldsymbol{n} \cdot (\boldsymbol{s}_2 - \boldsymbol{s}_1)}{d_{12}} > \frac{d_{12}}{2}\right] \tag{4.2–35}$$

or

$$P_b = \mathrm{P}\left[\boldsymbol{n} \cdot (\boldsymbol{s}_2 - \boldsymbol{s}_1) > \frac{d_{12}^2}{2}\right] \tag{4.2–36}$$

We note that $\boldsymbol{n} \cdot (\boldsymbol{s}_2 - \boldsymbol{s}_1)$ is a zero-mean Gaussian random variable with variance $\frac{d_{12}^2 N_0}{2}$; therefore, using Equation 2.3–12, we obtain

$$\begin{aligned} P_b &= Q\left(\frac{\frac{d_{12}^2}{2}}{d_{12}\sqrt{\frac{N_0}{2}}}\right) \\ &= Q\left(\sqrt{\frac{d_{12}^2}{2N_0}}\right) \end{aligned} \tag{4.2–37}$$

Equation 4.2–37 is very general and applies to all binary equiprobable signaling systems regardless of the shape of the signals. Since $Q(\cdot)$ is a decreasing function, in order to minimize the error probability, the distance between signal points has to be maximized. The distance d_{12} is obtained from

$$d_{12}^2 = \int_{-\infty}^{\infty} (s_1(t) - s_2(t))^2 \, dt \tag{4.2–38}$$

In the special case that the binary signals are equiprobable and have equal energy, i.e., when $\mathcal{E}_{s_1} = \mathcal{E}_{s_2} = \mathcal{E}$, we can expand Equation 4.2–38 and get

$$d_{12}^2 = \mathcal{E}_{s_1} + \mathcal{E}_{s_2} - 2\langle s_1(t), s_2(t)\rangle = 2\mathcal{E}(1 - \rho) \tag{4.2–39}$$

where ρ is the cross-correlation coefficient between $s_1(t)$ and $s_2(t)$ defined in Equation 2.1–25. Since $-1 \leq \rho \leq 1$, we observe from Equation 4.2–39 that the binary signals are maximally separated when $\rho = -1$, i.e., when the signals are antipodal. In this case the error probability of the system is minimized.

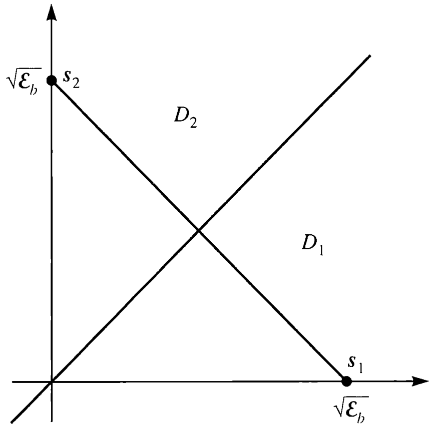


FIGURE 4.2–4
Signal constellation and decision regions for equiprobable binary orthogonal signaling.

Optimal Detection for Binary Orthogonal Signaling

For binary orthogonal signals we have

$$\int_{-\infty}^{\infty} s_i(t)s_j(t)\,dt = \begin{cases} \mathcal{E} & i = j \\ 0 & i \neq j \end{cases} \qquad 1 \leq i, j \leq 2 \tag{4.2–40}$$

Note that since the system is binary, $\mathcal{E}_b = \mathcal{E}$. Here we choose $\phi_j(t) = \frac{s_j(t)}{\sqrt{\mathcal{E}_b}}$ for $j = 1, 2$, and the vector representations of the signal set become

$$\begin{aligned} s_1 &= \left(\sqrt{\mathcal{E}_b}, 0\right) \\ s_2 &= \left(0, \sqrt{\mathcal{E}_b}\right) \end{aligned} \tag{4.2–41}$$

The constellation and the optimal decision regions for the case of equiprobable signals are shown in Figure 4.2–4.

For this signaling scheme it is clear that $d = \sqrt{2\mathcal{E}_b}$ and

$$P_b = Q\left(\sqrt{\frac{d^2}{2N_0}}\right) = Q\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right) \tag{4.2–42}$$

Comparing this result with the error probability of binary antipodal signaling given in Equation 4.2–34, we see that a binary orthogonal signaling requires twice the energy per bit of a binary antipodal signaling system to provide the same error probability. Therefore in terms of power efficiency, binary orthogonal signaling underperforms binary antipodal signaling by a factor of 2, or equivalently by 3 dB.

The term

$$\gamma_b = \frac{\mathcal{E}_b}{N_0} \tag{4.2–43}$$

which appears in the expression for error probability of many signaling systems is called the *signal-to-noise ratio per bit*, or *SNR per bit*, or simply the *SNR* of the communication system. Plots of error probability as a function of SNR/bit for binary antipodal and binary orthogonal signaling are shown in Figure 4.2–5. It is clear from this figure that the plot for orthogonal signaling is the result of a 3-dB shift of the plot for antipodal signaling.

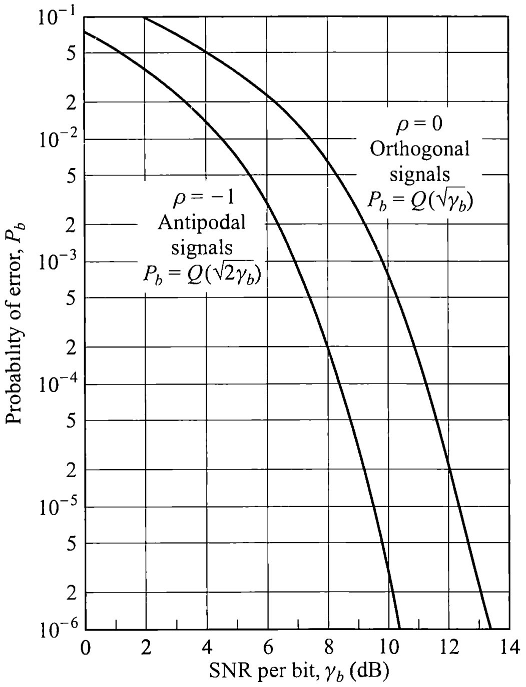


FIGURE 4.2–5
Error probability for binary antipodal and binary orthogonal signaling.

4.2–2 Implementation of the Optimal Receiver for AWGN Channels

In this section we present different implementations of the optimal (MAP) receiver for the AWGN channel. All these structures are equivalent in performance and result in minimum error probability. The underlying relation that is implemented by all these structures is Equation 4.2–17 which describes the MAP receiver for an AWGN channel.

The Correlation Receiver

An optimal receiver for the AWGN channel implements the MAP decision rule given by Equation 4.2–44.

$$\hat{m} = \underset{1\le m\le M}{\arg\max}\,[\eta_m + \boldsymbol{r}\cdot \boldsymbol{s}_m], \qquad \text{where } \eta_m = \frac{N_0}{2}\ln P_m - \frac{1}{2}\mathcal{E}_m \tag{4.2–44}$$

However, the receiver has access to $r(t)$ and not the vector $\boldsymbol{r}$. The first step to implement Equation 4.2–44 at the receiver is to derive $\boldsymbol{r}$ from the received signal $r(t)$. Using the relation

$$r_j = \int_{-\infty}^{\infty} r(t)\,\phi_j(t)\,dt \tag{4.2–45}$$

the receiver multiplies $r(t)$ by each basis function $\phi_j(t)$ and integrates the result to find all components of $\boldsymbol{r}$. In the next step it finds the inner product of $\boldsymbol{r}$ with each $\boldsymbol{s}_m$, $1 \le m \le M$, and finally adds the bias terms η_m and compares the results and chooses the m that maximizes the result. Since the received signal $r(t)$ is correlated with each $\phi_j(t)$, this implementation of the optimal receiver is called a *correlation receiver*.

The structure of a correlation receiver is shown in Figure 4.2–6.

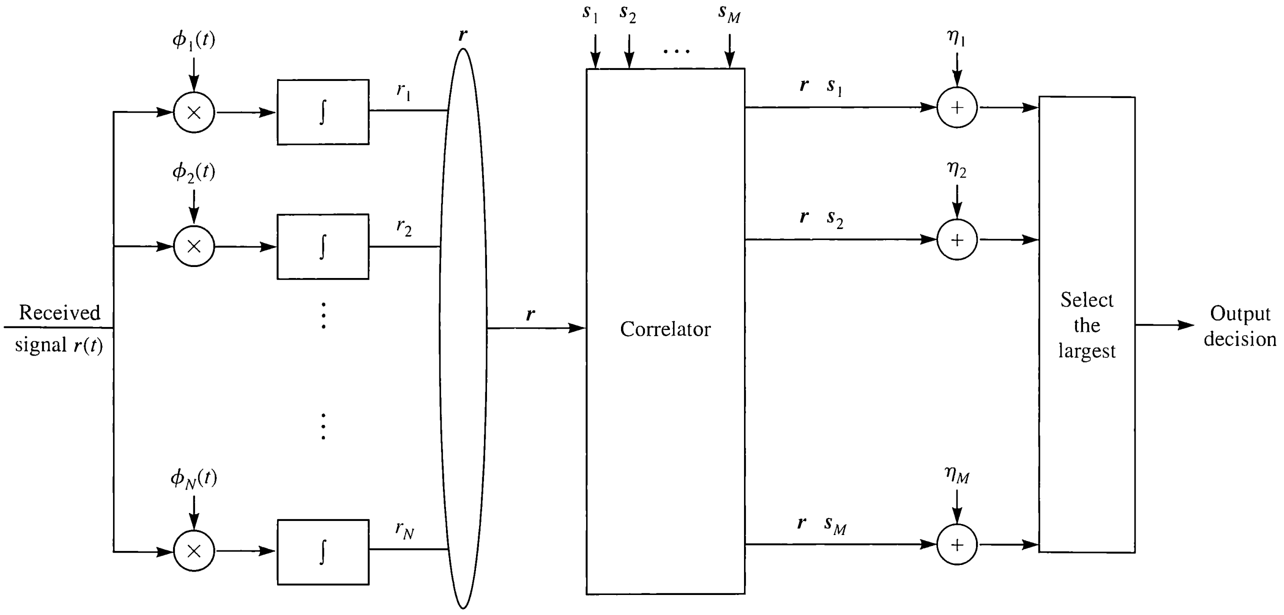


FIGURE 4.2–6
The structure of a correlation receiver with N correlators.

Note that in Figure 4.2–6, η_m's and s_m's are independent of the received signal $r(t)$; therefore they can be computed once and stored in a memory for later access. The parts of this diagram that need constant computation are the correlators that compute $\boldsymbol{r} \cdot \boldsymbol{s}_m$ for $1 \leq m \leq M$.

Another implementation of the optimal detector is possible by noting that the optimal detection rule given in Equation 4.2–44 is equivalent to

$$\hat{m} = \underset{1 \leq m \leq M}{\arg\max} \left[\eta_m + \int_{-\infty}^{\infty} r(t) s_m(t)\, dt \right], \qquad \text{where } \eta_m = \frac{N_0}{2} \ln P_m - \frac{1}{2}\mathcal{E}_m \tag{4.2–46}$$

Therefore, $\boldsymbol{r} \cdot \boldsymbol{s}_m$ can be directly found by correlation $r(t)$ with $s_m(t)$'s. Figure 4.2–7 shows this implementation which is a second version of the correlation receiver.

Note that although the structure shown in Figure 4.2–7 looks simpler than the structure shown in Figure 4.2–6, since in most cases $N < M$ (and in fact $N \ll M$), the correlation receiver of Figure 4.2–6 is usually the preferred implementation method.

The correlation receiver requires N or M correlators, i.e., multipliers followed by integrators. We now present an alternative implementation of the optimal receiver called the *matched filter receiver.*

The Matched Filter Receiver

In both correlation receiver implementations we compute quantities of the form

$$r_x = \int_{-\infty}^{\infty} r(t) x(t)\, dt \tag{4.2–47}$$

where $x(t)$ is either $\phi_j(t)$ or $s_m(t)$. If we define $h(t) = x(T - t)$, where T is arbitrary, and consider a filter with impulse response $h(t)$, this filter is called a *filter matched to*

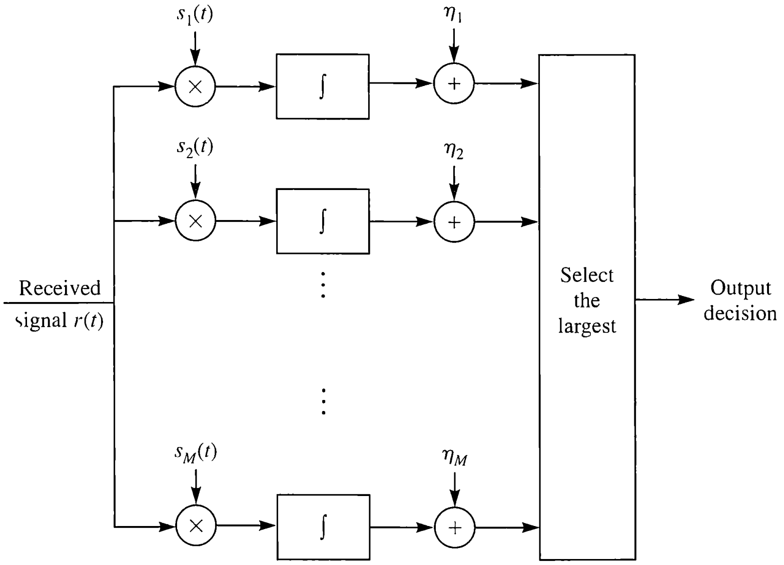


FIGURE 4.2–7
The structure of the correlation receiver with M correlators.

$x(t)$, or a matched filter. If the input $r(t)$ is applied to this filter, its output, denoted by $y(t)$, is the convolution of $r(t)$ and $h(t)$ and is given by

$$\begin{aligned} y(t) &= r(t) \star h(t) \\ &= \int_{-\infty}^{\infty} r(\tau)h(t-\tau)\,d\tau \\ &= \int_{-\infty}^{\infty} r(\tau)x(T-t+\tau)\,d\tau \end{aligned} \tag{4.2–48}$$

From Equation 4.2–48 it is clear that

$$r_x = y(T) = \int_{-\infty}^{\infty} r(\tau)x(\tau)\,d\tau \tag{4.2–49}$$

In other words, the output of the correlator r_x can be obtained by sampling the output of the matched filter at time $t = T$. Note that the sampling has to be done exactly at time $t = T$, where T is the arbitrary value used in the design of the matched filter. As long as this condition is satisfied, the choice of T is irrelevant; however from a practical point of view, T has to be selected in such a way that the resulting filters are causal; i.e, we must have $h(t) = 0$ for $t < 0$. This puts a practical limit on possible values of T. A matched filter implementation of the optimal receiver is shown in Figure 4.2–8.

Another matched filter implementation with M filters matched to $\{s_m(t), 1 \leq m \leq M\}$ similar to the correlation receiver shown in Figure 4.2–7 is also possible.

Frequency Domain Interpretation of the Matched Filter The matched filter to any signal $s(t)$ has an interesting frequency-domain interpretation. Since $h(t) = s(T-t)$, the Fourier transform of this relationship, using the basic properties of the Fourier

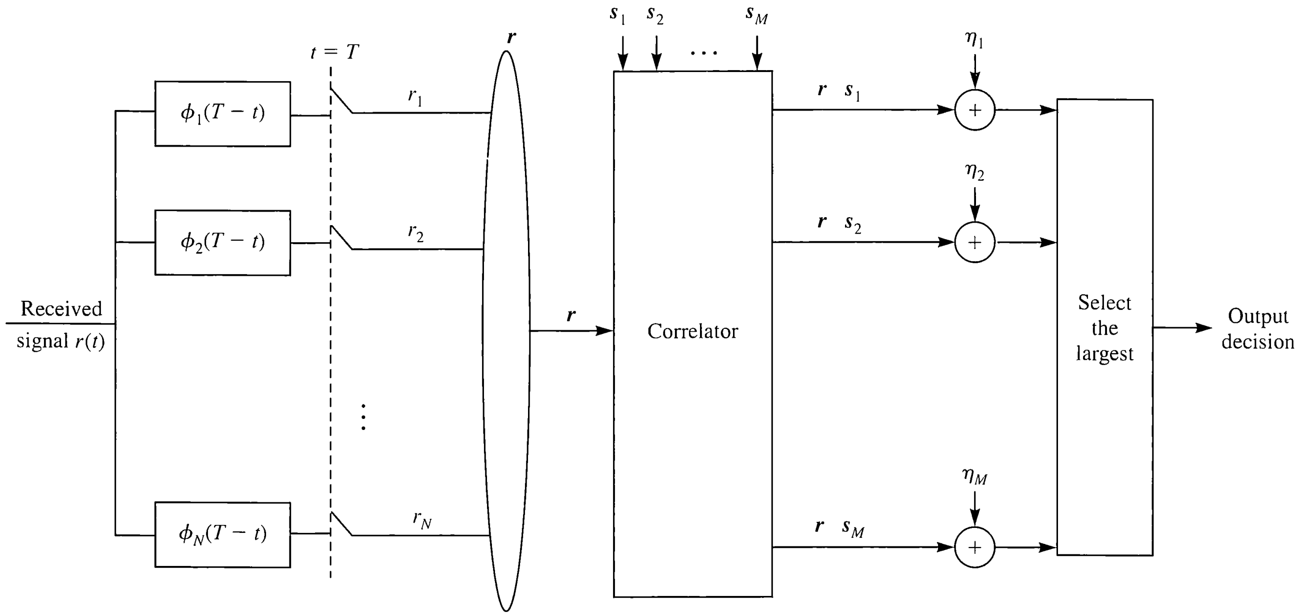

$t = T$
$\phi_1(T-t)$
$\phi_2(T-t)$
$\phi_N(T-t)$
r_1
r_2
r_N
Received signal $r(t)$
$\mathbf{r}$
$\mathbf{r}$
$\mathbf{s}_1$ $\mathbf{s}_2$ … $\mathbf{s}_M$
Correlator
$\mathbf{r}\cdot\mathbf{s}_1$
$\mathbf{r}\cdot\mathbf{s}_2$
$\mathbf{r}\cdot\mathbf{s}_M$
η_1
η_2
η_M
Select the largest
Output decision

FIGURE 4.2–8
The structure of a matched filter receiver with N correlators.

transform, is

$$H(f) = S^*(f)e^{-j2\pi fT} \tag{4.2–50}$$

We observe that the matched filter has a frequency response that is the complex conjugate of the transmitted signal spectrum multiplied by the phase factor $e^{-j2\pi fT}$, which represents the sampling delay of T. In other words, $|H(f)| = |S(f)|$, so that the magnitude response of the matched filter is identical to the transmitted signal spectrum. On the other hand, the phase of $H(f)$ is the negative of the phase of $S(f)$ shifted by $2\pi fT$.

Another interesting property of the matched filter is its signal-to-noise maximizing property. Let us assume that $r(t) = s(t) + n(t)$ is passed through a filter with impulse response $h(t)$ and frequency response $H(f)$, and the output, denoted by $y(t) = y_s(t) + \nu(t)$, is sampled at some time T. The output consists of a signal part, $y_s(t)$, whose Fourier transform is $H(f)S(f)$ and a noise part, $\nu(t)$, whose power spectral density is $\frac{N_0}{2}|H(f)|^2$. Sampling these components at time T results in

$$y_s(T) = \int_{-\infty}^{\infty} H(f)S(f)e^{j2\pi ft}\, dt \tag{4.2–51}$$

and a zero-mean Gaussian noise component, $\nu(T)$, whose variance is

$$\mathrm{VAR}\,[\nu(T)] = \frac{N_0}{2}\int_{-\infty}^{\infty} |H(f)|^2\, df = \frac{N_0}{2}\mathcal{E}_h \tag{4.2–52}$$

where $\mathcal{E}_h$ is the energy in $h(t)$. Now let us define the SNR at the output of the filter $H(f)$ as

$$\mathrm{SNR_o} = \frac{y_s^2(T)}{\mathrm{VAR}\,[\nu(T)]} \tag{4.2–53}$$

From the Cauchy-Schwartz inequality given in Equation 2.2–19, we have

$$\begin{aligned} y_s(T) &= \int_{-\infty}^{\infty} H(f)S(f)e^{j2\pi ft}\,dt \\ &\leq \int_{-\infty}^{\infty} |H(f)|^2\,df \cdot \int_{-\infty}^{\infty} |S(f)e^{j2\pi fT}|^2\,df \\ &= \mathcal{E}_h\,\mathcal{E}_s \end{aligned} \tag{4.2–54}$$

with equality if and only if $H(f) = \alpha S^*(f)e^{-j2\pi fT}$ for some complex constant α. Using Equation 4.2–54 in 4.2–53, we conclude that

$$\text{SNR}_o \leq \frac{\mathcal{E}_s\mathcal{E}_h}{\frac{N_0}{2}\mathcal{E}_h} = \frac{2\mathcal{E}_s}{N_0} \tag{4.2–55}$$

This shows that the filter $H(f)$ that maximizes the signal-to-noise ratio at its output must satisfy the relation $H(f) = S^*(f)e^{-j2\pi fT}$; i.e., it is the matched filter. It also shows that the maximum possible signal-to-noise ratio at the output is $\frac{2\mathcal{E}_s}{N_0}$.

EXAMPLE 4.2–1. $M = 4$ biorthogonal signals are constructed from the two orthogonal signals shown in Figure 4.2–9(a) for transmitting information over an AWGN channel. The noise is assumed to have zero mean and power spectral density $\frac{1}{2}N_0$. Let us determine the basis functions for this signal set, the impulse responses of the matched filter demodulators, and the output waveforms of the matched filter demodulators when the transmitted signal is $s_1(t)$.

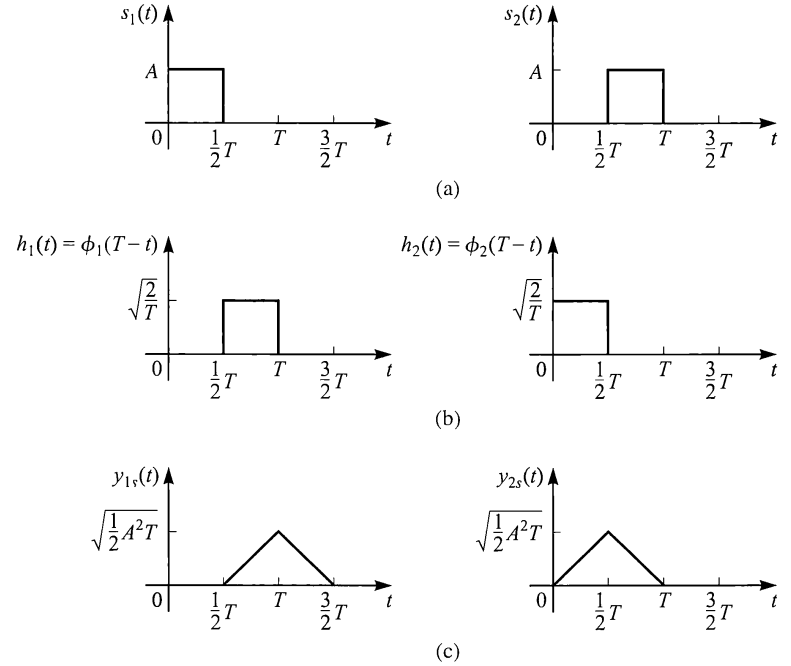

$s_1(t)$
A
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
$s_2(t)$
A
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
(a)
$h_1(t) = \phi_1(T-t)$
$\sqrt{\frac{2}{T}}$
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
$h_2(t) = \phi_2(T-t)$
$\sqrt{\frac{2}{T}}$
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
(b)
$y_{1s}(t)$
$\sqrt{\frac{1}{2}A^2T}$
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
$y_{2s}(t)$
$\sqrt{\frac{1}{2}A^2T}$
0
$\frac{1}{2}T$
T
$\frac{3}{2}T$
t
(c)

FIGURE 4.2–9
Basis functions and matched filter response for Example 4.2–1.

The $M = 4$ biorthogonal signals have dimensions $N = 2$. Hence, two basis functions are needed to represent the signals. From Figure 4.2–9(a), we choose $\phi_1(t)$ and $\phi_2(t)$ as

$$\begin{aligned} \phi_1(t) &= \begin{cases} \sqrt{2/T} & 0 \le t \le \frac{1}{2}T \\ 0 & \text{otherwise} \end{cases} \\ \phi_2(t) &= \begin{cases} \sqrt{2/T} & \frac{1}{2}T \le t \le T \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4.2–56}$$

The impulse responses of the two matched filters are

$$\begin{aligned} h_1(t) = \phi_1(T - t) &= \begin{cases} \sqrt{2/T} & \frac{1}{2}T \le t \le T \\ 0 & \text{otherwise} \end{cases} \\ h_2(t) = \phi_2(T - t) &= \begin{cases} \sqrt{2/T} & 0 \le t \le \frac{1}{2}T \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{4.2–57}$$

and are illustrated in Figure 4.2–9(b).

If $s_1(t)$ is transmitted, the (noise-free) responses of the two matched filters are as shown in Figure 4.2–9(c). Since $y_1(t)$ and $y_2(t)$ are sampled at $t = T$, we observe that $y_{1s}(T) = \sqrt{\frac{1}{2}A^2T}$ and $y_{2s}(T) = 0$. Note that $\frac{1}{2}A^2T = \mathcal{E}$, the signal energy. Hence, the received vector formed from the two matched filter outputs at the sampling instant $t = T$ is

$$\boldsymbol{r} = (r_1, r_2) = (\sqrt{\mathcal{E}} + n_1, n_2) \tag{4.2–58}$$

where $n_1 = y_{1n}(T)$ and $n_2 = y_{2n}(T)$ are the noise components at the outputs of the matched filters, given by

$$y_{kn}(T) = \int_0^T n(t)\phi_k(t)\,dt, \qquad k = 1, 2 \tag{4.2–59}$$

Clearly, $\mathrm{E}[n_k] = \mathrm{E}[y_{kn}(T)] = 0$. Their variance from Equation 4.2–52 is

$$\mathrm{VAR}[n_k] = \frac{N_0}{2}\mathcal{E}_{\phi_k} = \frac{1}{2}N_0 \tag{4.2–60}$$

Observe that the SNR for the first matched filter is

$$\mathrm{SNR}_o = \frac{(\sqrt{\mathcal{E}})^2}{\frac{1}{2}N_0} = \frac{2\mathcal{E}}{N_0} \tag{4.2–61}$$

which agrees with our previous result.

4.2–3 A Union Bound on the Probability of Error of Maximum Likelihood Detection

In general, to determine the error probability of a signaling scheme, we need to use Equation 4.1–13. In the special case where the messages are equiprobable, $P_m = 1/M$

and maximum likelihood detection is optimal. The error probability in this case becomes

$$\begin{aligned} P_e &= \frac{1}{M}\sum_{m=1}^{M} P_{e|m} \\ &= \frac{1}{M}\sum_{m=1}^{M} \sum_{\substack{1\le m'\le M \\ m'\ne m}} \int_{D_{m'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \end{aligned} \tag{4.2–62}$$

For an AWGN channel the decision regions are given by Equation 4.2–20. Therefore, for AWGN channels we have

$$\begin{aligned} P_{e|m} &= \sum_{\substack{1\le m'\le M \\ m'\ne m}} \int_{D_{m'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \\ &= \sum_{\substack{1\le m'\le M \\ m'\ne m}} \int_{D_{m'}} p_{\boldsymbol{n}}(\boldsymbol{r}-\boldsymbol{s}_m)\,d\boldsymbol{r} \\ &= \left(\frac{1}{\sqrt{\pi N_0}}\right)^N \sum_{\substack{1\le m'\le M \\ m'\ne m}} \int_{D_{m'}} e^{-\frac{\|\boldsymbol{r}-\boldsymbol{s}_m\|^2}{N_0}}\,d\boldsymbol{r} \end{aligned} \tag{4.2–63}$$

For very few constellations, decision regions $D_{m'}$ are regular enough that the integrals in the last line of Equation 4.2–63 or Equation 4.2–62 can be computed in a closed form. For most constellations (for example, look at Figure 4.2–1) these integrals cannot be put in a closed form. In such cases it is convenient to have upper bounds for the error probability. There exist many bounds on the error probability under ML detection. The union bound is the simplest and most widely used bound which is quite tight particularly at high signal-to-noise ratios.

We first derive the union bound for a general communication channel and then study the AWGN channel as a special case. First we note that in general the decision region $D_{m'}$ under ML detection can be expressed as

$$D_{m'} = \left\{\boldsymbol{r}\in\mathbb{R}^N : p(\boldsymbol{r}|\boldsymbol{s}_{m'}) > p(\boldsymbol{r}|\boldsymbol{s}_k),\quad \text{for all } 1\le k\le M \text{ and } k\ne m'\right\} \tag{4.2–64}$$

Let us define $D_{mm'}$ as

$$D_{mm'} = \{p(\boldsymbol{r}|\boldsymbol{s}_{m'}) > p(\boldsymbol{r}|\boldsymbol{s}_m)\} \tag{4.2–65}$$

Note that $D_{mm'}$ is the decision region for m' in a binary equiprobable system with signals $\boldsymbol{s}_m$ and $\boldsymbol{s}_{m'}$. Comparing the definitions of $D_{m'}$ and $D_{mm'}$, we obviously have

$$D_{m'} \subseteq D_{mm'} \tag{4.2–66}$$

hence

$$\int_{D_{m'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \le \int_{D_{mm'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \tag{4.2–67}$$

Note that the right-hand side of this equation is the error probability of a binary equiprobable system with signals $\boldsymbol{s}_m$ and $\boldsymbol{s}_{m'}$ when $\boldsymbol{s}_m$ is transmitted. We define the *pairwise error probability*, denoted by $P_{m \to m'}$ as

$$P_{m \to m'} = \int_{D_{mm'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\, d\boldsymbol{r} \tag{4.2–68}$$

From Equations 4.2–63 and 4.2–67 we have

$$\begin{aligned} P_{e|m} &\le \sum_{\substack{1 \le m' \le M \\ m' \ne m}} \int_{D_{mm'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\, d\boldsymbol{r} \\ &= \sum_{\substack{1 \le m' \le M \\ m' \ne m}} P_{m \to m'} \end{aligned} \tag{4.2–69}$$

and from Equation 4.2–62 we conclude that

$$\begin{aligned} P_e &\le \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{1 \le m' \le M \\ m' \ne m}} \int_{D_{mm'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\, d\boldsymbol{r} \\ &= \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{1 \le m' \le M \\ m' \ne m}} P_{m \to m'} \end{aligned} \tag{4.2–70}$$

Equations 4.2–70 is the union bound for a general communication channel.

In the special case of an AWGN channel, we know from Equation 4.2–37 that the pairwise error probability is given by

$$P_{m \to m'} = P_b = Q\left(\sqrt{\frac{d_{mm'}^2}{2N_0}}\right) \tag{4.2–71}$$

By using this result, Equation 4.2–70 becomes

$$\begin{aligned} P_e &\le \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{1 \le m' \le M \\ m' \ne m}} Q\left(\sqrt{\frac{d_{mm'}^2}{2N_0}}\right) \\ &\le \frac{1}{2M} \sum_{m=1}^{M} \sum_{\substack{1 \le m' \le M \\ m' \ne m}} e^{-\frac{d_{mm'}^2}{4N_0}} \end{aligned} \tag{4.2–72}$$

where in the last step we have used the upper bound on the Q function given in Equation 2.3–15 as

$$Q(x) \le \frac{1}{2} e^{-\frac{x^2}{2}} \tag{4.2–73}$$

Equation 4.2–72 is the general form of the union bound for an AWGN channel. If we know the distance structure of the constellation, we can further simplify this bound.

Let us define $T(X)$, the *distance enumerator function* for a constellation, as

$$\begin{aligned} T(X) &= \sum_{\substack{d_{mm'}=\|s_m-s_{m'}\| \\ 1\le m,m'\le M \\ m\neq m'}} X^{d^2_{mm'}} \\ &= \sum_{\text{all distinct } d\text{'s}} a_d X^{d^2} \end{aligned} \tag{4.2–74}$$

where a_d denotes the number of ordered pairs (m, m') such that $m \neq m'$ and $\|s_m - s_{m'}\| = d$. Using this function, Equation 4.2–72 can be written as

$$P_e \le \frac{1}{2M} T(X)\Big|_{X=e^{-\frac{1}{4N_0}}} \tag{4.2–75}$$

Let us define $d_{\min}$, the *minimum distance* of a constellation, as

$$d_{\min} = \min_{\substack{1\le m,m'\le M \\ m\neq m'}} \|s_m - s_{m'}\| \tag{4.2–76}$$

Since $Q(\cdot)$ is decreasing, we have

$$Q\left(\sqrt{\frac{d^2_{mm'}}{2N_0}}\right) \le Q\left(\sqrt{\frac{d^2_{\min}}{2N_0}}\right) \tag{4.2–77}$$

Substituting in Equation 4.2–70 results in

$$P_e \le (M-1)Q\left(\sqrt{\frac{d^2_{\min}}{2N_0}}\right) \tag{4.2–78}$$

Equation 4.2–78 is a looser form of the union bound in terms of the Q function and $d_{\min}$ which has a very simple form. Using the exponential bound for the Q function we have the union bound in the simple form

$$P_e \le \frac{M-1}{2} e^{-\frac{d^2_{\min}}{4N_0}} \tag{4.2–79}$$

The union bound clearly shows that the minimum distance of a constellation has an important impact on the performance of the communication system. A good constellation should be designed such that, within the power and bandwidth constraints, it provides the maximum possible minimum distance; i.e., the points in the constellation should be maximally separated.

EXAMPLE 4.2–2. Let us consider the 16-QAM constellation shown in Figure 4.2–10. We assume that the distance between any two adjacent points on the constellation is $d_{\min}$. From Equation 3.2–44 we have

$$d_{\min} = \sqrt{\frac{6\log_2 M}{M-1}\mathcal{E}_{\text{bavg}}} = \sqrt{\frac{8}{5}\mathcal{E}_{\text{bavg}}} \tag{4.2–80}$$

Close observation of this constellation shows that from a total of $16 \times 15 = 240$ possible distances between any two points in the constellation, 48 are equal to $d_{\min}$, 36 are equal to $\sqrt{2}\,d_{\min}$, 32 are $2d_{\min}$, 48 are $\sqrt{5}\,d_{\min}$, 16 are $\sqrt{8}\,d_{\min}$, 16 are $3d_{\min}$, 24 are $\sqrt{10}\,d_{\min}$, 16 are $\sqrt{13}\,d_{\min}$, and finally 4 are $\sqrt{18}\,d_{\min}$. Note that each line connecting

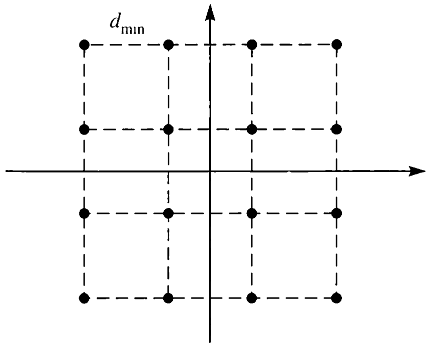


FIGURE 4.2–10
16-QAM constellation.

any two points in the constellation is counted twice. Therefore, the distance enumerator function for this constellation is given by

$$T(X) = 48X^{d^2} + 36X^{2d^2} + 32X^{4d^2} + 48X^{5d^2} + 16X^{8d^2} + 16X^{9d^2} + 24X^{10d^2} + 16X^{13d^2} + 4X^{18d^2} \tag{4.2–81}$$

where for ease of notation we have substituted $d_{\min}$ by d. The union bound becomes

$$P_e \leq \frac{1}{32} T\left(e^{-\frac{1}{4N_0}}\right) \tag{4.2–82}$$

A looser, but simpler, form of the union bound is obtained in terms of $d_{\min}$ as

$$P_e \leq \frac{M-1}{2} e^{-\frac{d_{\min}^2}{4N_0}} = \frac{15}{2} e^{-\frac{2\mathcal{E}_{\text{bavg}}}{5N_0}} \tag{4.2–83}$$

where in the last step we have used Equation 4.2–80.

In the case when $d_{\min}^2$ is large compared to N_0, i.e., when SNR is large, the first term is the dominating term in Equation 4.2–82. In this case we have

$$P_e \lessapprox \frac{48}{32} e^{-\frac{d_{\min}^2}{4N_0}} = \frac{3}{2} e^{-\frac{2\mathcal{E}_{\text{bavg}}}{5N_0}} \tag{4.2–84}$$

It turns out that for this constellation it is possible to derive an exact expression for the error probability (see Example 4.3–1), and the expression for the error probability is given by

$$P_e = 3Q\left(\sqrt{\frac{4\mathcal{E}_{\text{bavg}}}{5N_0}}\right) - \frac{9}{4}\left[Q\left(\sqrt{\frac{4\mathcal{E}_{\text{bavg}}}{5N_0}}\right)\right]^2 \tag{4.2–85}$$

Plots of the exact error probability and the upper bounds given by Equations 4.2–83 and 4.2–84 are shown in Figure 4.2–11.

A Lower Bound on the Probability of Error

In an equiprobable M-ary signaling scheme, the error probability is given by

$$\begin{aligned} P_e &= \frac{1}{M} \sum_{m=1}^{M} \mathrm{P}\left[\text{Error} \mid m \text{ sent}\right] \\ &= \frac{1}{M} \sum_{m=1}^{M} \int_{D_m^c} p(\boldsymbol{r} \mid \boldsymbol{s}_m)\, d\boldsymbol{r} \end{aligned} \tag{4.2–86}$$

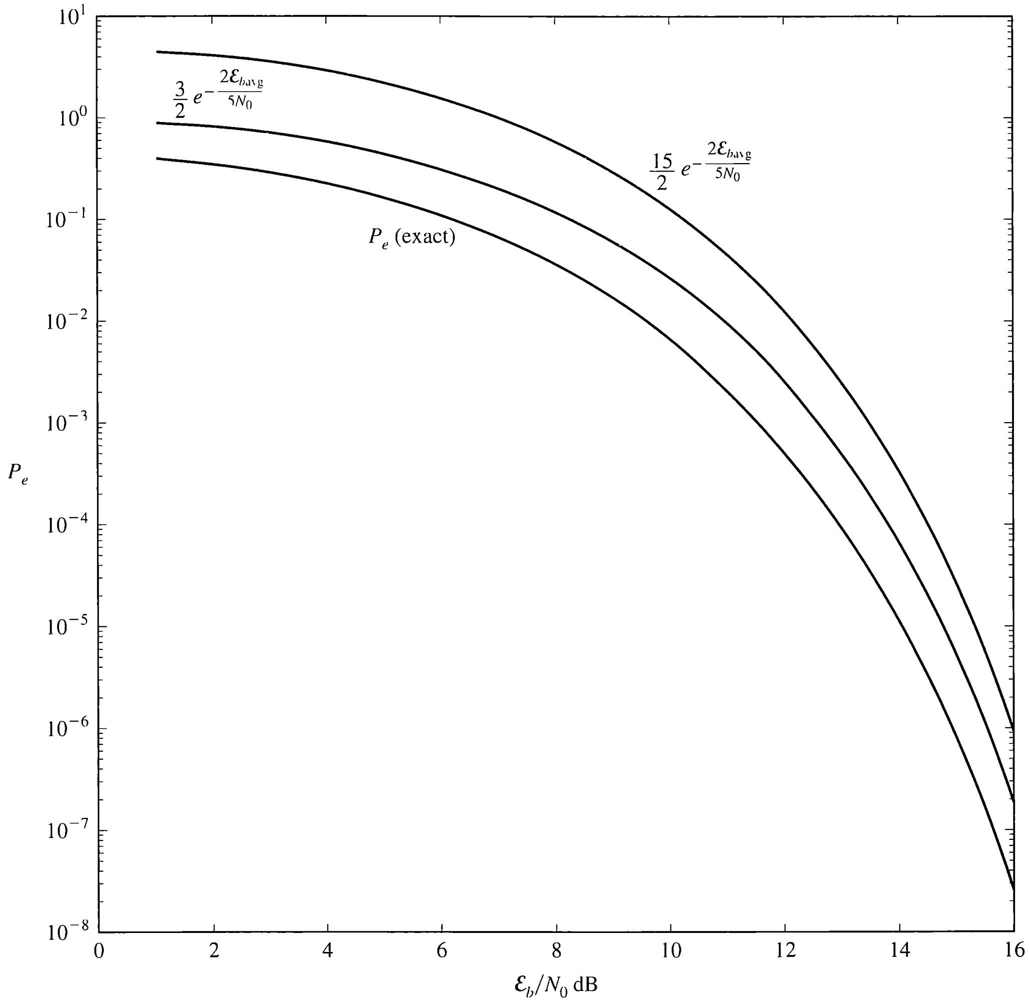


FIGURE 4.2–11
Comparison of the exact error probability and two upper bounds for rectangular 16-QAM.

From Equation 4.2–66 we have $D^c_{m'm} \subseteq D^c_m$; hence,

$$\begin{aligned} P_e &\geq \frac{1}{M}\sum_{m=1}^{M}\int_{D^c_{m'm}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \\ &= \frac{1}{M}\sum_{m=1}^{M}\int_{D_{mm'}} p(\boldsymbol{r}|\boldsymbol{s}_m)\,d\boldsymbol{r} \\ &= \frac{1}{M}\sum_{m=1}^{M} Q\left(\frac{d_{mm'}}{\sqrt{2N_0}}\right) \end{aligned} \tag{4.2–87}$$

Equation 4.2–87 is valid for all $m' \neq m$. To derive the tightest lower bound, we need to maximize the right-hand side. Therefore we can write

$$P_e \geq \frac{1}{M}\sum_{m=1}^{M} \max_{m' \neq m} Q\left(\frac{d_{mm'}}{\sqrt{2N_0}}\right) \tag{4.2–88}$$

Since the Q function is a decreasing function of its variable, choosing m' that maximizes $Q\left(\frac{d_{mm'}}{\sqrt{2N_0}}\right)$ is equivalent to finding m' such that $d_{mm'}$ is minimized. Hence,

$$P_e \geq \frac{1}{M} \sum_{m=1}^{M} Q\left(\frac{d_{\min}^m}{\sqrt{2N_0}}\right) \tag{4.2–89}$$

where $d_{\min}^m$ denotes the distance from m to its nearest neighbor in the constellation, and obviously $d_{\min}^m \geq d_{\min}$. Therefore,

$$Q\left(\frac{d_{\min}^m}{\sqrt{2N_0}}\right) \geq \begin{cases} Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) & \text{if there exists at least one signal at distance } d_{\min} \text{ from } \boldsymbol{s}_m \\ 0 & \text{otherwise} \end{cases} \tag{4.2–90}$$

By using Equation 4.2–90, Equation 4.2–89 becomes

$$P_e \geq \frac{1}{M} \sum_{\substack{1 \leq m \leq M \\ \exists m' \neq m \; \|\boldsymbol{s}_m - \boldsymbol{s}_{m'}\| = d_{\min}}} Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.2–91}$$

Denoting by $N_{\min}$ the number of the points in the constellation that are at the distance from $d_{\min}$ from at least one other point in the constellation, we obtain

$$P_e \geq \frac{N_{\min}}{M} Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.2–92}$$

From Equations 4.2–92 and 4.2–78, it is clear that

$$\frac{N_{\min}}{M} Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \leq P_e \leq (M-1) Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.2–93}$$

4.3

OPTIMAL DETECTION AND ERROR PROBABILITY FOR BAND-LIMITED SIGNALING

In this section we study signaling schemes that are mainly characterized by their low bandwidth requirements. These signaling schemes have low dimensionality which is independent from the number of transmitted signals, and, as we will see, their power efficiency decreases when the number of messages increases. This family of signaling schemes includes ASK, PSK, and QAM.

4.3–1 Optimal Detection and Error Probability for ASK or PAM Signaling

The constellation for an ASK signaling scheme is shown in Figure 4.3–1. In this constellation the minimum distance between any two points is $d_{\min}$ which is given by Equation 3.2–22 as

$$d_{\min} = \sqrt{\frac{12 \log_2 M}{M^2 - 1} \mathcal{E}_{\text{bavg}}} \tag{4.3–1}$$

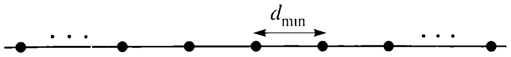


FIGURE 4.3–1
The ASK constellation.

The constellation points are located at $\left\{\pm\frac{1}{2}d_{\min}, \pm\frac{3}{2}d_{\min}, \dots, \pm\frac{M-1}{2}d_{\min}\right\}$.

We notice there exist two types of points in the ASK constellation. There are $M-2$ inner points and 2 outer points in the constellation. If an inner point is transmitted, there will be an error in detection if $|n| > \frac{1}{2}d_{\min}$. For the outer points, the probability of error is one-half of the error probability of an inner point since noise can cause error in only one direction. Let us denote the error probabilities of inner points and outer points by P_{ei} and P_{eo}, respectively. Since n is a zero-mean Gaussian random variable with variance $\frac{1}{2}N_0$, we have

$$P_{ei} = \mathrm{P}\left[|n| > \frac{1}{2}d_{\min}\right] = 2Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.3–2}$$

and for the outer points

$$P_{eo} = \frac{1}{2}P_{ei} = Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.3–3}$$

The symbol error probability is given by

$$\begin{aligned} P_e &= \frac{1}{M}\sum_{m=1}^{M} \mathrm{P}\left[\text{error}\,|\,m \text{ sent}\right] \\ &= \frac{1}{M}\left[2(M-2)Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) + 2Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right)\right] \\ &= \frac{2(M-1)}{M}Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \end{aligned} \tag{4.3–4}$$

Substituting for $d_{\min}$ from Equation 4.3–1yields

$$\begin{aligned} P_e &= 2\left(1-\frac{1}{M}\right)Q\left(\sqrt{\frac{6\log_2 M}{M^2-1}\frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \\ &\approx 2Q\left(\sqrt{\frac{6\log_2 M}{M^2-1}\frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \qquad \text{for large } M \end{aligned} \tag{4.3–5}$$

Note that the average SNR/bit $\frac{\mathcal{E}_{\text{bavg}}}{N_0}$ is scaled by $\frac{6\log_2 M}{M^2-1}$. This factor goes to 0 as M increases, which means that to keep the error probability constant as M increases, the SNR/bit must increase. For large M, doubling M—which is equivalent to increasing the transmission rate by 1 bit per transmission—would roughly need the SNR/bit to quadruple, i.e., an increase of 6 dB, to keep the performance the same. In other words, as a rule of thumb, for increasing the transmission rate by 1 bit, one would need 6 dB more power.

Plots of the error probability of baseband PAM and ASK as a function of the average SNR/bit for different values of M are given in Figure 4.3–2. It is clear that

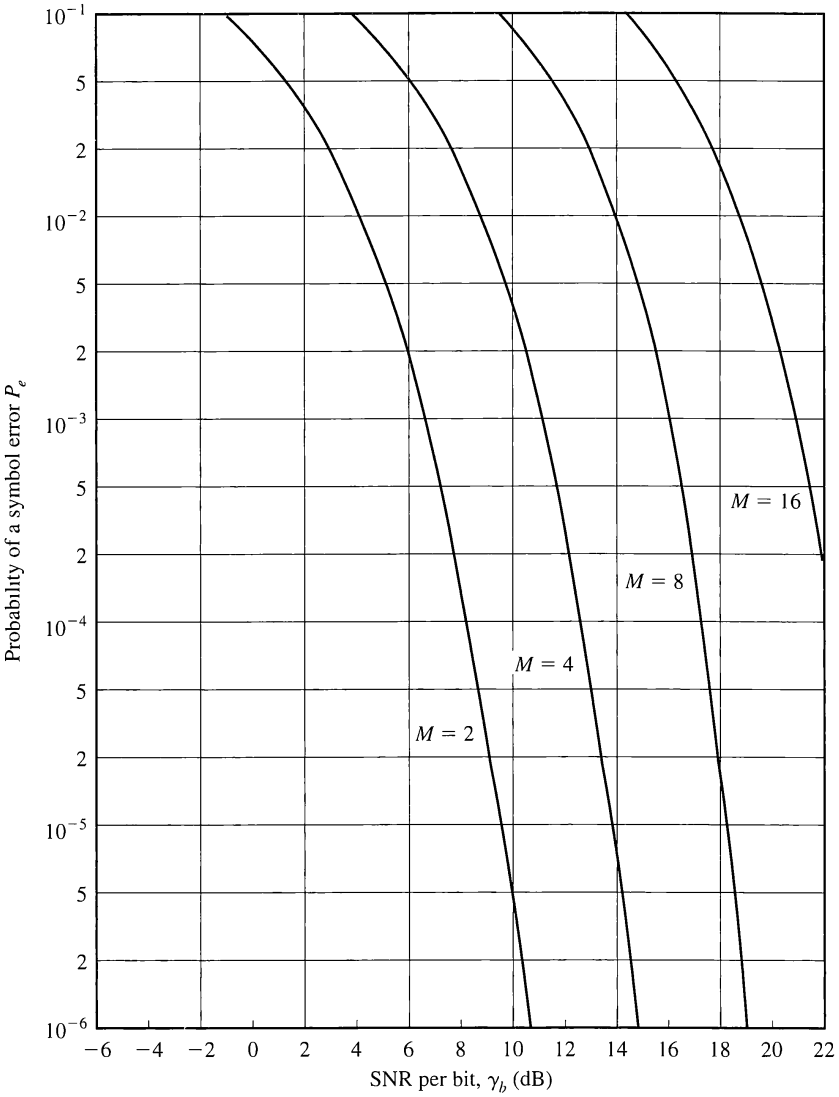


FIGURE 4.3–2
Symbol error probability for baseband PAM and ASK.

increasing M deteriorates the performance, and for large M the distance between curves corresponding to M and $2M$ is roughly 6 dB.

4.3–2 Optimal Detection and Error Probability for PSK Signaling

The constellation for an M-ary PSK signaling is shown in Figure 4.3–3. In this constellation the decision region D_1 is also shown. Note that since we are assuming the messages are equiprobable, the decision regions are based on the minimum-distance detection rule. By symmetry of the constellation, the error probability of the system is equal to the error probability when $s_1 = (\sqrt{\mathcal{E}}, 0)$ is transmitted. The received vector $\boldsymbol{r}$ is given by

$$\boldsymbol{r} = (r_1, r_2) = (\sqrt{\mathcal{E}} + n_1, n_2) \tag{4.3–6}$$

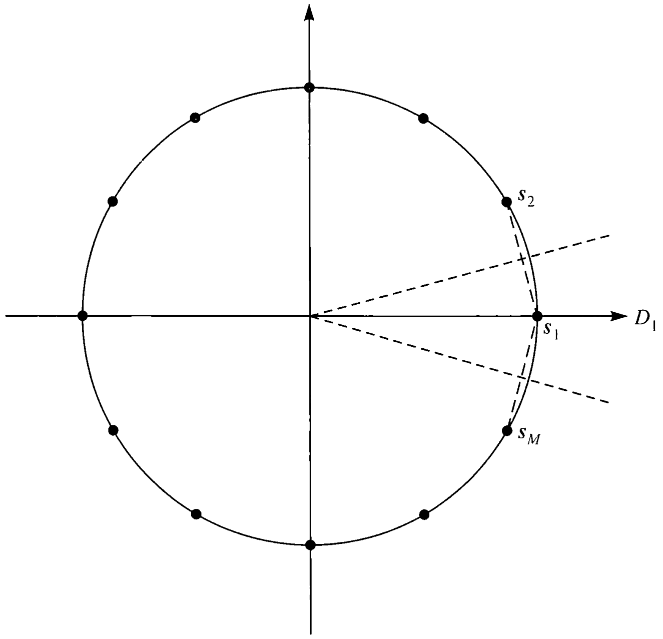


FIGURE 4.3–3
The constellation for PSK signaling.

It is seen that r_1 and r_2 are independent Gaussian random variables with variance $\sigma^2 = \frac{1}{2}N_0$ and means $\sqrt{\mathcal{E}}$ and 0, respectively; hence

$$p(r_1, r_2) = \frac{1}{\pi N_0} e^{-\frac{(r_1-\sqrt{\mathcal{E}})^2+r_2^2}{N_0}} \tag{4.3–7}$$

Since the decision region D_1 can be more conveniently described using polar coordinates, we introduce polar coordinates transformations of (r_1, r_2) as

$$\begin{aligned} V &= \sqrt{r_1^2 + r_2^2} \\ \Theta &= \arctan \frac{r_2}{r_1} \end{aligned} \tag{4.3–8}$$

from which the joint PDF of V and Θ can be derived as

$$p_{V,\Theta}(v, \theta) = \frac{v}{\pi N_0} e^{-\frac{v^2+\mathcal{E}-2\sqrt{\mathcal{E}}\, v \cos\theta}{N_0}} \tag{4.3–9}$$

Integrating over v, we derive the marginal PDF of Θ as

$$\begin{aligned} p_\Theta(\theta) &= \int_0^\infty p_{V,\Theta}(v, \theta)\, dv \\ &= \frac{1}{2\pi} e^{-\gamma_s \sin^2\theta} \int_0^\infty v\, e^{-\frac{\left(v-\sqrt{2\gamma_s}\cos\theta\right)^2}{2}}\, dv \end{aligned} \tag{4.3–10}$$

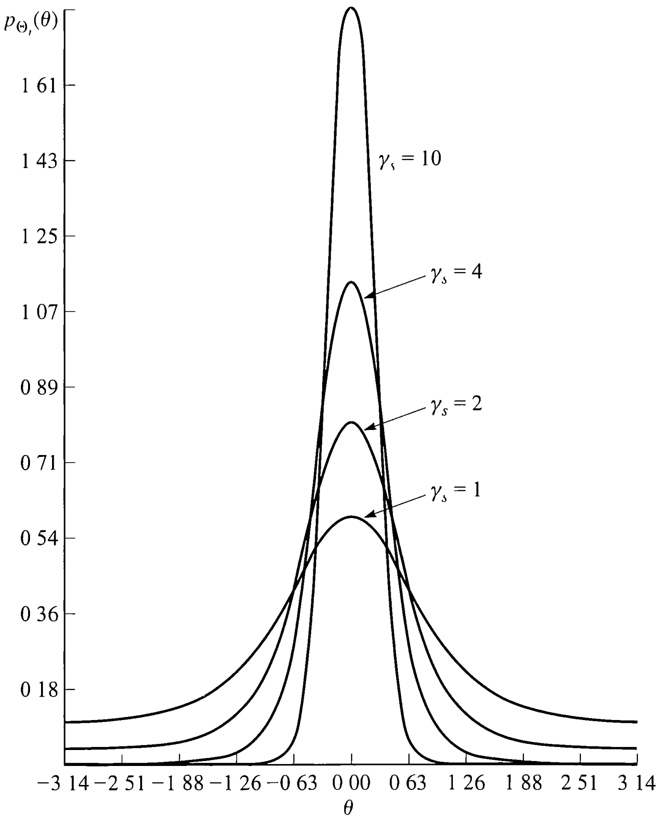


FIGURE 4.3–4
The PDF of Θ for $\gamma_s = 1, 2, 4$, and 10.

in which we have defined the *symbol SNR* or *SNR per symbol* as

$$\gamma_s = \frac{\mathcal{E}}{N_0} \tag{4.3–11}$$

Figure 4.3–4 illustrates $p_{\Theta}(\theta)$ for several values of γ_s. Note that $p_{\Theta}(\theta)$ becomes narrower and more peaked about $\theta = 0$ as γ_s increases.

The decision region D_1 can be described as $D_1 = \{\theta : -\pi/M < \theta \leq \pi/M\}$; therefore, the message error probability is given by

$$P_e = 1 - \int_{-\pi/M}^{\pi/M} p_{\Theta}(\theta)\, d\theta \tag{4.3–12}$$

In general, the integral of $p_{\Theta}(\theta)$ does not reduce to a simple form and must be evaluated numerically, except for $M = 2$ and $M = 4$.

For binary phase modulation, the two signals $s_1(t)$ and $s_2(t)$ are antipodal, and hence the error probability is

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{4.3–13}$$

When $M = 4$, we have in effect two binary phase-modulation signals in phase quadrature. Since there is no crosstalk or interference between the signals on the two quadrature carriers, the bit error probability is identical to that in Equation 4.3–13. On the other hand, the symbol error probability for $M = 4$ is determined by noting that

$$P_c = (1 - P_b)^2 = \left[1 - Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\right]^2 \tag{4.3–14}$$

where P_c is the probability of a correct decision for the 2-bit symbol. Equation 4.3–14 follows from the statistical independence of the noise on the quadrature carriers. Therefore, the symbol error probability for $M = 4$ is

$$\begin{aligned} P_e &= 1 - P_c \\ &= 2Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\right] \end{aligned} \tag{4.3–15}$$

For $M > 4$, the symbol error probability P_e is obtained by numerically integrating Equation 4.3–12. Figure 4.3–5 illustrates this error probability as a function of the SNR per bit for $M = 2, 4, 8, 16$, and 32. The graphs clearly illustrate the penalty in SNR per bit as M increases beyond $M = 4$. For example, at $P_e = 10^{-5}$, the difference between $M = 4$ and $M = 8$ is approximately 4 dB, and the difference between $M = 8$ and $M = 16$ is approximately 5 dB. For large values of M, doubling the number of phases

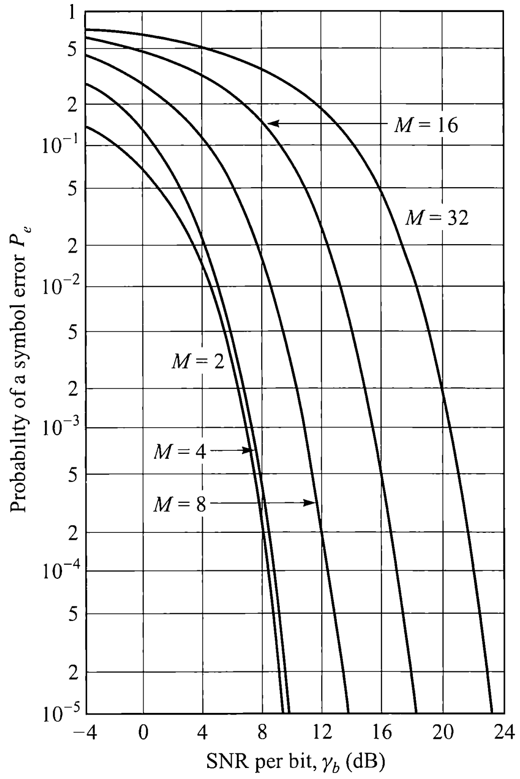


FIGURE 4.3–5
Probability of symbol error for PSK signals.

requires an additional 6 dB/bit to achieve the same performance. This performance is similar to the performance of ASK signaling discussed in Section 4.3–1.

An approximation to the error probability for large values of M and for large SNR may be obtained by first approximating $p_\Theta(\theta)$. For $\mathcal{E}/N_0 \gg 1$ and $|\theta| \le \frac{1}{2}\pi$, $p_\Theta(\theta)$ is well approximated as

$$p_\Theta(\theta) \approx \sqrt{\frac{\gamma_s}{\pi}} \cos\theta \, e^{-\gamma_s \sin^2\theta} \tag{4.3–16}$$

By substituting for $p_\Theta(\theta)$ in Equation 4.3–12 and performing the change in variable from θ to $u = \sqrt{\gamma_s} \sin\theta$, we find that

$$\begin{aligned} P_e &\approx 1 - \int_{-\pi/M}^{\pi/M} \sqrt{\frac{\gamma_s}{\pi}} \cos\theta \, e^{-\gamma_s \sin^2\theta} \, d\theta \\ &\approx \frac{2}{\sqrt{\pi}} \int_{\sqrt{2\gamma_s}\sin(\pi/M)}^{\infty} e^{-u^2} \, du \\ &= 2Q\left(\sqrt{2\gamma_s} \sin\left(\frac{\pi}{M}\right)\right) \\ &= 2Q\left(\sqrt{(2\log_2 M)\sin^2\left(\frac{\pi}{M}\right)\frac{\mathcal{E}_b}{N_0}}\right) \end{aligned} \tag{4.3–17}$$

where we have used the definition of the SNR per bit as

$$\frac{\mathcal{E}_b}{N_0} = \frac{\mathcal{E}}{N_0 \log_2 M} = \frac{\gamma_s}{\log_2 M} \tag{4.3–18}$$

Note that this approximation[†] to the error probability is good for all values of M. For example, when $M = 2$ and $M = 4$, we have $P_e = 2Q(\sqrt{2\gamma_b})$ which compares favorably with the exact probabilities given by Equations 4.3–13 and 4.3–15.

For the case when M is large, we can use the approximation $\sin\frac{\pi}{M} \approx \frac{\pi}{M}$ to find another approximation to error probability for large M as

$$P_e \approx 2Q\left(\sqrt{\frac{2\pi^2 \log_2 M}{M^2} \frac{\mathcal{E}_b}{N_0}}\right) \qquad \text{for large } M \tag{4.3–19}$$

From Equation 4.3–19 it is clear that doubling M reduces the effective SNR per bit by 6 dB.

The equivalent bit error probability for M-ary PSK is rather tedious to derive due to its dependence on the mapping of k-bit symbols into the corresponding signal phases. When a Gray code is used in the mapping, two k-bit symbols corresponding to adjacent signal phases differ in only a single bit. Since the most probable errors due to noise result in the erroneous selection of an adjacent phase to the true phase, most k-bit symbol errors contain only a single-bit error. Hence, the equivalent bit error probability for M-ary PSK is well approximated as

---

[†] A better approximation of the error probability at low SNR is given in the paper by Lu et al (1999)

$$P_b \approx \frac{1}{k} P_e \tag{4.3–20}$$

Differentially Encoded PSK Signaling

Our treatment of the demodulation of PSK signals assumed that the demodulator had a perfect estimate of the carrier phase available. In practice, however, the carrier phase is extracted from the received signal by performing some nonlinear operation that introduces a phase ambiguity. For example, in binary PSK, the signal is often squared in order to remove the modulation, and the double-frequency component that is generated is filtered and divided by 2 in frequency in order to extract an estimate of the carrier frequency and phase ϕ. These operations result in a phase ambiguity of 180° in the carrier phase. Similarly, in four-phase PSK, the received signal is raised to the fourth power to remove the digital modulation, and the resulting fourth harmonic of the carrier frequency is filtered and divided by 4 to extract the carrier component. These operations yield a carrier frequency component containing the estimate of the carrier phase ϕ, but there are phase ambiguities of $\pm 90^\circ$ and 180° in the phase estimate. Consequently, we do not have an absolute estimate of the carrier phase for demodulation.

The phase ambiguity problem resulting from the estimation of the carrier phase ϕ can be overcome by encoding the information in phase differences between successive signal transmissions as opposed to absolute phase encoding. For example, in binary PSK, the information bit 1 may be transmitted by shifting the phase of the carrier by 180° relative to the previous carrier phase, while the information bit 0 is transmitted by a zero phase shift relative to the phase in the previous signaling interval. In four-phase PSK, the relative phase shifts between successive intervals are 0°, 90°, 180°, and -90°, corresponding to the information bits 00, 01, 11, and 10, respectively. The generalization to M phases is straightforward. The PSK signals resulting from the encoding process are said to be *differentially encoded*. The encoding is performed by a relatively simple logic circuit preceding the modulator.

Demodulation of the differentially encoded PSK signal is performed as described above, by ignoring the phase ambiguities. Thus, the received signal is demodulated and detected to one of the M possible transmitted phases in each signaling interval. Following the detector is a relatively simple phase comparator that compares the phases of the demodulated signal over two consecutive intervals to extract the information.

Coherent demodulation of differentially encoded PSK results in a higher probability of error than the error probability derived for absolute phase encoding. With differentially encoded PSK, an error in the demodulated phase of the signal in any given interval will usually result in decoding errors over two consecutive signaling intervals. This is especially the case for error probabilities below 0.1. Therefore, the probability of error in differentially encoded M-ary PSK is approximately twice the probability of error for M-ary PSK with absolute phase encoding. However, this factor-of-2 increase in the error probability translates into a relatively small loss in SNR.

4.3–3 Optimal Detection and Error Probability for QAM Signaling

In optimal detection of QAM signals, we need two filters matched to

$$\begin{aligned}\phi_1(t) &= \sqrt{\frac{2}{E_g}}\, g(t)\cos 2\pi f_c t \\ \phi_2(t) &= -\sqrt{\frac{2}{E_g}}\, g(t)\sin 2\pi f_c t\end{aligned} \tag{4.3–21}$$

The output of the matched filters $\boldsymbol{r} = (r_1, r_2)$ is used to compute $C(\boldsymbol{r}, \boldsymbol{s}_m) = 2\boldsymbol{r} \cdot \boldsymbol{s}_m - \mathcal{E}_m$, and the largest is selected. The resulting decision regions depend on the constellation shape, and in general the error probability does not have a closed form.

To determine the probability of error for QAM, we must specify the signal point constellation. We begin with QAM signal sets that have $M = 4$ points. Figure 4.3–6 illustrates two four-point signal sets. The first is a four-phase modulated signal, and the second is a QAM signal with two amplitude levels, labeled A_1 and A_2, and four phases. Because the probability of error is dominated by the minimum distance between pairs of signal points, let us impose the condition that $d_{\min} = 2A$ for both signal constellations and let us evaluate the average transmitter power, based on the premise that all signal points are equally probable. For the four-phase signal, we have

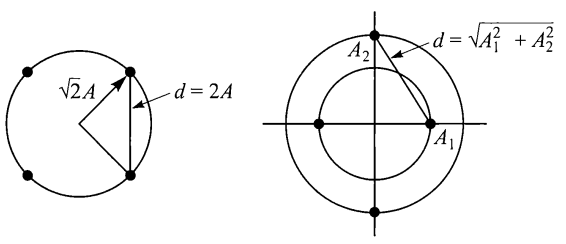
$$\mathcal{E}_{\text{avg}} = 2A^2 \tag{4.3–22}$$

For the two-amplitude, four-phase QAM, we place the points on circles of radii A and $\sqrt{3}A$. Thus, $d_{\min} = 2A$, and

$$\mathcal{E}_{\text{avg}} = \frac{1}{4}\left[2(3A^2) + 2A^2\right] = 2A^2 \tag{4.3–23}$$

which is the same average power as the $M = 4$-phase signal constellation. Hence, for all practical purposes, the error rate performance of the two signal sets is the same. In other words, there is no advantage of the two-amplitude QAM signal set over $M = 4$-phase modulation.

Next, let us consider $M = 8$-QAM. In this case, there are many possible signal constellations. We shall consider the four signal constellations shown in Figure 4.3–7, all of which consist of two amplitudes and have a minimum distance between signal points of $2A$. The coordinates (A_{mc}, A_{ms}) for each signal point, normalized by A, are given in the figure. Assuming that the signal points are equally probable, the average



(a) (b)

FIGURE 4.3–6
Two four-point signal constellations.

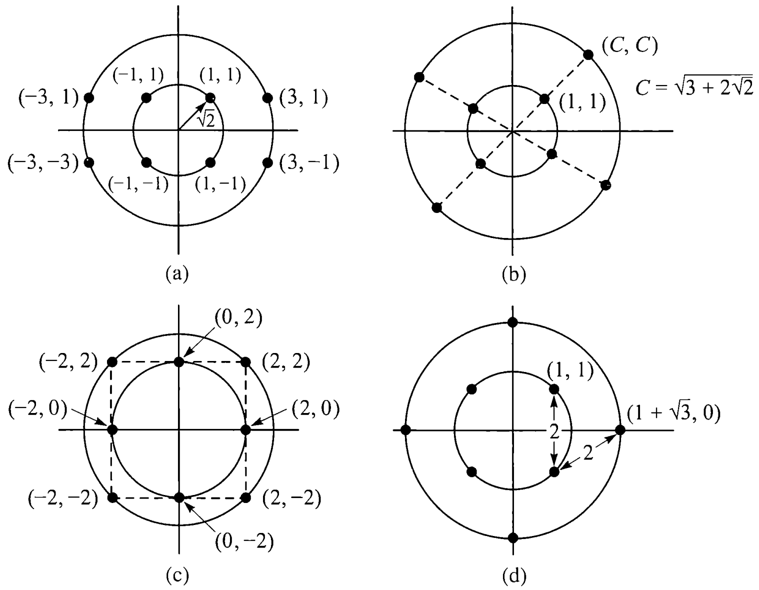


FIGURE 4.3–7
Four eight-point signal constellations.

transmitted signal energy is

$$\begin{aligned}\mathcal{E}_{\text{avg}} &= \frac{1}{M}\sum_{m=1}^{M}\left(A_{mc}^2 + A_{ms}^2\right)\\ &= \frac{A^2}{M}\sum_{m=1}^{M}\left(a_{mc}^2 + a_{ms}^2\right)\end{aligned} \tag{4.3–24}$$

where (a_{mc}, a_{ms}) are the coordinates of the signal points, normalized by A.

The two signal sets (a) and (c) in Figure 4.3–7 contain signal points that fall on a rectangular grid and have $\mathcal{E}_{\text{avg}} = 6A^2$. The signal set (b) requires an average transmitted energy $\mathcal{E}_{\text{avg}} = 6.83A^2$, and (d) requires $\mathcal{E}_{\text{avg}} = 4.73A^2$. Therefore, the fourth signal set requires approximately 1 dB less energy than the first two and 1.6 dB less energy than the third, to achieve the same probability of error. This signal constellation is known to be the best eight-point QAM constellation because it requires the least power for a given minimum distance between signal points.

For $M \geq 16$, there are many more possibilities for selecting the QAM signal points in two-dimensional space. For example, we may choose a circular multiamplitude constellation for $M = 16$, as shown in Figure 3.2–4. In this case, the signal points at a given amplitude level are phase-rotated by $\frac{1}{4}\pi$ relative to the signal points at adjacent amplitude levels. This 16-QAM constellation is a generalization of the optimum 8-QAM constellation. However, the circular 16-QAM constellation is not the best 16-point QAM signal constellation for the AWGN channel.

Rectangular QAM signal constellations have the distinct advantage of being easily generated as two PAM signals impressed on the in-phase and quadrature carriers. In addition, they are easily demodulated. Although they are not the best M-ary QAM signal constellations for $M \geq 16$, the average transmitted power required to achieve a given minimum distance is only slightly greater than the average power required for the best M-ary QAM signal constellation. For these reasons, rectangular M-ary QAM signals are most frequently used in practice.

In the special case where k is even and the constellation is square, it is possible to derive an exact expression for the error probability. This particular case was previously studied in Section 3.2–3 in Equations 3.2–42 to 3.2–44. In particular, the minimum distance of this constellation is given by

$$d_{\min} = \sqrt{\frac{6 \log_2 M}{M-1} \mathcal{E}_{\text{bavg}}} \tag{4.3–25}$$

Note that this constellation can be considered as two $\sqrt{M}$-ary PAM constellations in the in-phase and quadrature directions. An error occurs if either n_1 or n_2 is large enough to cause an error in one of the two PAM signals. The probability of a correct detection for this QAM constellation is therefore the product of correct decision probabilities for constituent PAM systems, i.e.,

$$P_{c,M\text{-QAM}} = P^2_{c,\sqrt{M}\text{-PAM}} = \left(1 - P_{e,\sqrt{M}\text{-PAM}}\right)^2 \tag{4.3–26}$$

resulting in

$$\begin{aligned} P_{e,M\text{-QAM}} &= 1 - \left(1 - P_{e,\sqrt{M}\text{-PAM}}\right)^2 \\ &= 2P_{e,\sqrt{M}\text{-PAM}} \left(1 - \frac{1}{2} P_{e,\sqrt{M}\text{-PAM}}\right) \end{aligned} \tag{4.3–27}$$

But, from Equation 4.3–4,

$$P_{e,\sqrt{M}\text{-PAM}} = 2\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\frac{d_{\min}}{\sqrt{2N_0}}\right) \tag{4.3–28}$$

in which we need to substitute $d_{\min}$ from Equation 4.3–25 to obtain

$$P_{e,\sqrt{M}\text{-PAM}} = 2\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \tag{4.3–29}$$

Substituting Equation 4.3–29 into Equation 4.3–27 yields

$$\begin{aligned} P_{e,M\text{-QAM}} &= 4\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \\ &\quad \times \left(1 - \left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right)\right) \\ &\leq 4Q\left(\sqrt{\frac{3 \log_2 M}{M-1} \frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \end{aligned} \tag{4.3–30}$$

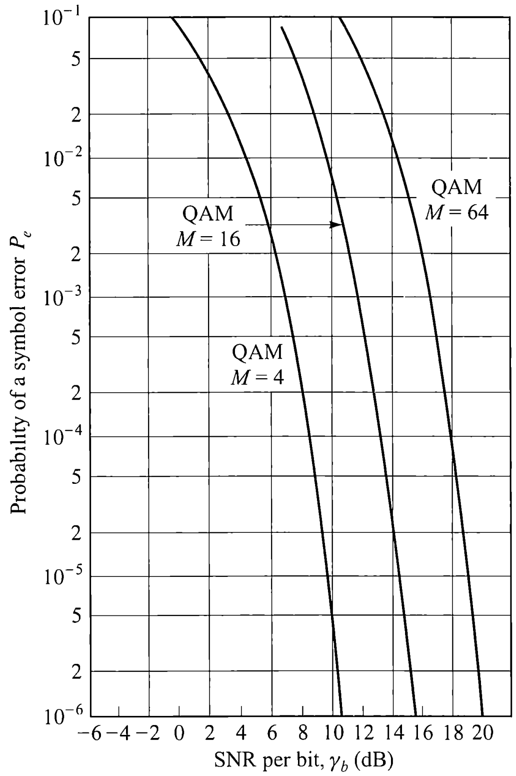


FIGURE 4.3–8
Probability of a symbol error for QAM.

For large M and moderate to high SNR per bit, the upper bound given in Equation 4.3–30 is quite tight. Figure 4.3–8 illustrates plots of message error probability of M-ary QAM as a function of SNR per bit. Although Equation 4.3–30 is obtained for square constellations, for large M it gives a good approximation for general QAM constellations with $M = 2^k$ points which are either in the shape of a square (when k is even) or in the shape of a cross (when k is odd). These types of constellations are illustrated in Figure 3.2–5.

Comparing the error performance of M-ary QAM with M-ary ASK and MPSK given in Equations 4.3–5 and 4.3–19, respectively, we observe that unlike PAM and PSK signaling in which in the penalty for increasing the rate was 6 dB/bit, in QAM this penalty is 3 dB/bit. This shows that QAM is more power efficient compared with PAM and PSK. The advantage of PSK is, however, its constant-envelope properties.

EXAMPLE 4.3–1. QPSK can be considered as 4-QAM with a square constellation. Using Equation 4.3–30 with $M = 4$, we obtain

$$\begin{aligned} P_4 &= 2Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\left[1 - \frac{1}{2}Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)\right] \\ &\le 2Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \end{aligned} \tag{4.3–31}$$

which is in agreement wit Equation 4.3–15. For 16-QAM with a rectangular constellation we obtain

$$\begin{aligned} P_{16} &= 3Q\left(\sqrt{\frac{4}{5}\frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right)\left[1-\frac{3}{4}Q\left(\sqrt{\frac{4}{5}\frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right)\right] \\ &\leq 3Q\left(\sqrt{\frac{4}{5}\frac{\mathcal{E}_{\text{bavg}}}{N_0}}\right) \end{aligned} \tag{4.3–32}$$

For nonrectangular QAM signal constellations, we may upper-bound the error probability by use of the union bound as

$$P_M \leq (M-1)Q\left(\sqrt{\frac{d_{\min}^2}{2N_0}}\right) \tag{4.3–33}$$

where $d_{\min}$ is the minimum Euclidean distance of the constellation. This bound may be loose when M is large. In such a case, we may approximate P_M by replacing $M-1$ by $N_{\min}$, where $N_{\min}$ is the largest number of neighboring points that are at distance $d_{\min}$ from any constellation point. More discussion on the performance of general QAM signaling schemes is given in Section 4.7.

It is interesting to compare the performance of QAM with that of PSK for any given signal size M, since both types of signals are two-dimensional. Recall that by Equation 4.3–17, for M-ary PSK, the probability of a symbol error is approximated as

$$P_M \approx 2Q\left(\sqrt{(2\log_2 M)\,\sin^2\left(\frac{\pi}{M}\right)\frac{\mathcal{E}_b}{N_0}}\right) \tag{4.3–34}$$

For M-ary QAM, we may use the expression 4.3–30. Since the error probability is dominated by the argument of the Q function, we may simply compare the arguments of Q for the two signal formats. Thus, the ratio of these two arguments is

$$\mathcal{R}_M = \frac{\frac{3}{M-1}}{2\sin^2\left(\frac{\pi}{M}\right)} \tag{4.3–35}$$

For example, when $M = 4$, we have $\mathcal{R}_M = 1$. Hence, 4-PSK and 4-QAM yield comparable performance for the same SNR per symbol. This was noted in Example 4.3–1. On the other hand, when $M > 4$, we find that $\mathcal{R}_M > 1$, so that M-ary QAM yields better performance than M-ary PSK. Table 4.3–1 illustrates the SNR advantage of QAM over PSK for several values of M. For example, we observe that 32-QAM has a 7-dB SNR advantage over 32-PSK.

TABLE 4.3–1
SNR Advantage of M-ary QAM over M-ary PSK

M	$10 \log \mathcal{R}_M$
8	1.65
16	4.20
32	7.02
64	9.95

4.3–4 Demodulation and Detection

ASK, PSK, and QAM have one- or two-dimensional constellations with orthonormal basis of the form

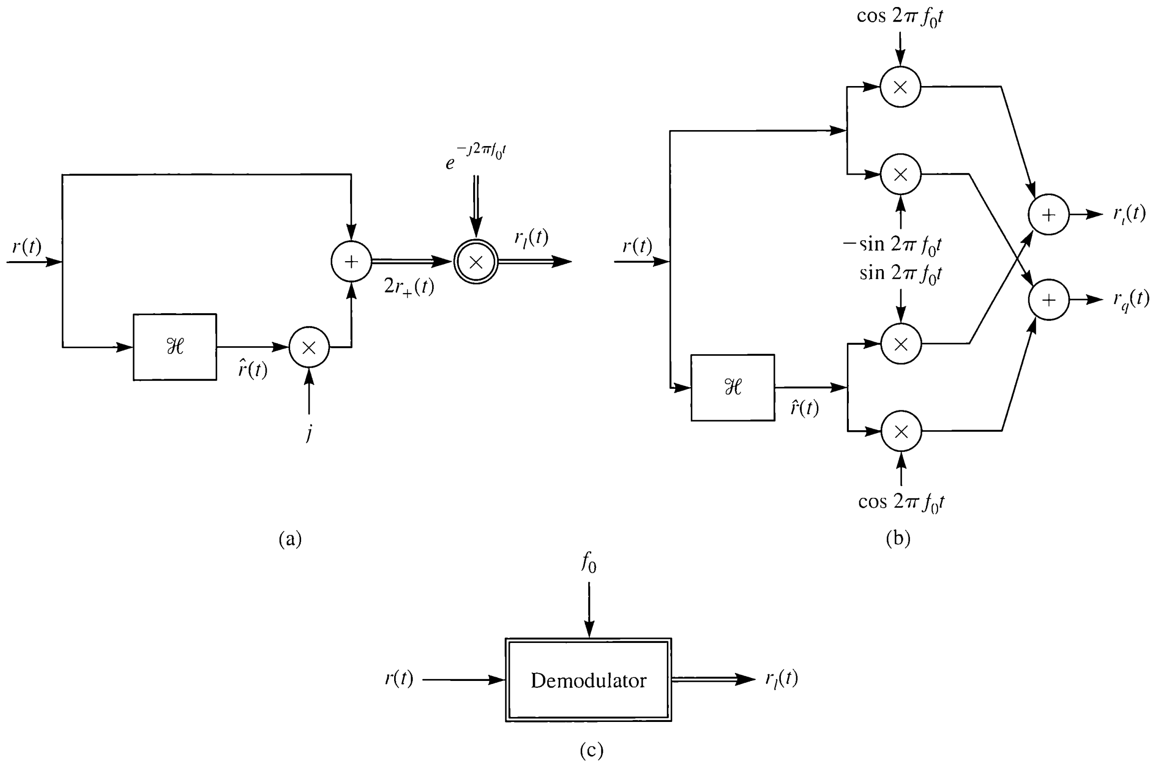
$$\begin{aligned}\phi_1(t) &= \sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\cos 2\pi f_c t \\ \phi_2(t) &= -\sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\sin 2\pi f_c t\end{aligned} \tag{4.3–36}$$

for PSK and QAM and

$$\phi_1(t) = \sqrt{\frac{2}{\mathcal{E}_g}}\, g(t)\cos 2\pi f_c t \tag{4.3–37}$$

for ASK. The optimal detector in these systems requires filters matched to $\phi_1(t)$ and $\phi_2(t)$. Since both the received signal $r(t)$ and the basis functions are high frequency bandpass signals, the filtering process, if implemented in software, requires high sampling rates.

To alleviate this requirement, we can first demodulate the received signal to obtain its lowpass equivalent signal and then perform the detection on this signal. The process of demodulation was previously discussed in Section 2.1–2 and the block diagram of the demodulator is repeated in Figure 4.3–9.



$r(t)$
$e^{-j2\pi f_0 t}$
$r_l(t)$
$2r_+(t)$
$\mathcal{H}$
$\hat{r}(t)$
j
(a)
$\cos 2\pi f_0 t$
$-\sin 2\pi f_0 t$
$\sin 2\pi f_0 t$
$\cos 2\pi f_0 t$
$r(t)$
$\mathcal{H}$
$\hat{r}(t)$
$r_i(t)$
$r_q(t)$
(b)
f_0
$r(t)$
Demodulator
$r_l(t)$
(c)

FIGURE 4.3–9

Complex (a) and real (b) demodulators. A general representation for a demodulator is shown in (c).

It is important to note that the demodulation process is an invertible process. We have seen in Section 4.1–1 that invertible preprocessing does not affect optimality of the receiver. Therefore, the optimal detector designed for the demodulated signal performs as well as the optimal detector designed for the bandpass signal. The benefit of the demodulator-detector implementation is that in this structure the signal processing required for the detection is done on the demodulated lowpass signal, thus reducing the complexity of the receiver.

Recall from Equations 2.1–21 and 2.1–24 that $\mathcal{E}_x = \frac{1}{2}\mathcal{E}_{x_l}$ and $\langle x(t), y(t)\rangle = \frac{1}{2}\operatorname{Re}\left[\langle x_l(t), y_l(t)\rangle\right]$. From these relations the optimal detection rule

$$\hat{m} = \arg\max_{1\le m\le M}\left(\boldsymbol{r}\cdot\boldsymbol{s}_m + \frac{N_0}{2}\ln P_m - \frac{1}{2}\mathcal{E}_m\right) \tag{4.3–38}$$

can be written in the following lowpass equivalent form

$$\hat{m} = \arg\max_{1\le m\le M}\left(\operatorname{Re}\left[\boldsymbol{r}_l\cdot\boldsymbol{s}_{ml}\right] + N_0\ln P_m - \frac{1}{2}\mathcal{E}_{ml}\right) \tag{4.3–39}$$

or, equivalently,

$$\hat{m} = \arg\max_{1\le m\le M}\left(\operatorname{Re}\left[\int_{-\infty}^{\infty} r_l(t)s_{ml}^*(t)\,dt\right] + N_0\ln P_m - \frac{1}{2}\int_{-\infty}^{\infty}|s_{ml}(t)|^2\,dt\right) \tag{4.3–40}$$

The ML detection rule is obviously

$$\hat{m} = \arg\max_{1\le m\le M}\left(\operatorname{Re}\left[\int_{-\infty}^{\infty} r_l(t)s_{ml}^*(t)\,dt\right] - \frac{1}{2}\int_{-\infty}^{\infty}|s_{ml}(t)|^2\,dt\right) \tag{4.3–41}$$

Equations 4.3–39 to 4.3–41 are baseband detection rules after demodulation.

The implementation of Equations 4.3–39 to 4.3–41 can be done either in the form of a correlation receiver or in the form of matched filters where the matched filters are of the form $s_{ml}^*(T-t)$ or $\phi_{jl}^*(T-t)$. Figure 4.3–10 shows the schematic diagram for a complex matched filter, and Figure 4.3–11 illustrates the detailed structure of a complex matched filter in terms of its in-phase and quadrature components. Note that for ASK, PSK, and QAM we have $s_{ml}(t) = A_m g(t)$, where A_m is in general a complex number (real for ASK). Therefore $\phi_1(t) = g(t)/\sqrt{\mathcal{E}_g}$ serves as the basis function, and the signal points are represented by complex numbers of the form $A_m\sqrt{\mathcal{E}_g}$. Also note that for PSK detection the last term in Equation 4.3–41 can be dropped.

Throughout this discussion we have assumed that the receiver has complete knowledge of the carrier frequency and phase. This requires full synchronization between the

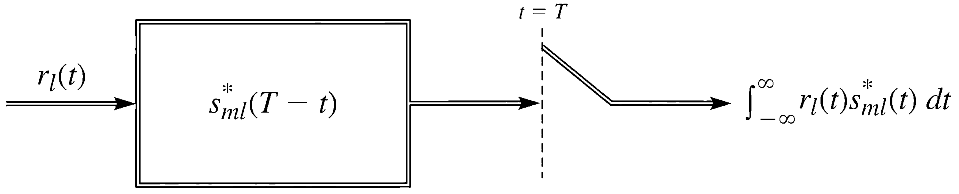


FIGURE 4.3–10
Complex lowpass equivalent matched filter.

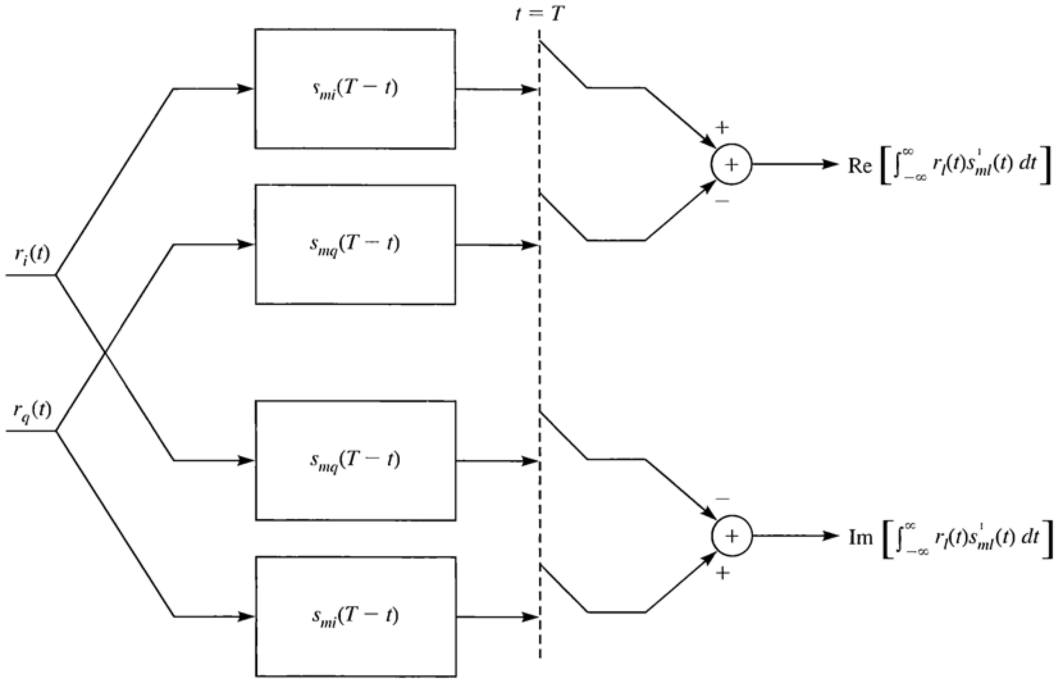

$t = T$
$s_{mi}(T-t)$
$s_{mq}(T-t)$
$s_{mq}(T-t)$
$s_{mi}(T-t)$
$r_i(t)$
$r_q(t)$
+
−
−
+
$\mathrm{Re}\left[\int_{-\infty}^{\infty} r_l(t)s_{ml}^*(t)\,dt\right]$
$\mathrm{Im}\left[\int_{-\infty}^{\infty} r_l(t)s_{ml}^*(t)\,dt\right]$

FIGURE 4.3–11
Equivalent lowpass matched filter.

transmitter and the receiver. In Section 4.5 we will study the case where the carrier generated at the receiver is not in phase coherence with the transmitter carrier.

4.4
OPTIMAL DETECTION AND ERROR PROBABILITY FOR POWER-LIMITED SIGNALING

Orthogonal, biorthogonal, and simplex signaling is characterized by high dimensional constellations. As we will see in this section, these signaling schemes are more power-efficient but less bandwidth-efficient than ASK, PSK, and QAM. We begin our study with orthogonal signaling and then extend our results to biorthogonal and simplex signals.

4.4–1 Optimal Detection and Error Probability for Orthogonal Signaling

In an equal-energy orthogonal signaling scheme, $N = M$ and the vector representation of the signals is given by

$$\begin{aligned} \boldsymbol{s}_1 &= (\sqrt{\mathcal{E}}, 0, \ldots, 0) \\ \boldsymbol{s}_2 &= (0, \sqrt{\mathcal{E}}, \ldots, 0) \\ \vdots &= \vdots \\ \boldsymbol{s}_M &= (0, \ldots, 0, \sqrt{\mathcal{E}}) \end{aligned} \tag{4.4–1}$$

For equiprobable, equal-energy orthogonal signals, the optimum detector selects the signal resulting in the largest cross-correlation between the received vector $\boldsymbol{r}$ and each of the M possible transmitted signal vectors $\{\boldsymbol{s}_m\}$, i.e.,

$$\hat{m} = \arg\max_{1 \le m \le M} \boldsymbol{r} \cdot \boldsymbol{s}_m \tag{4.4–2}$$

By symmetry of the constellation and by observing that the distance between any pair of signal points in the constellation is equal to $\sqrt{2\mathcal{E}}$, we conclude that the error probability is independent of the transmitted signal. Therefore, to evaluate the probability of error, we can suppose that the signal $\boldsymbol{s}_1$ is transmitted. With this assumption, the received signal vector is

$$\boldsymbol{r} = \left(\sqrt{\mathcal{E}} + n_1, n_2, n_3, \ldots, n_M\right) \tag{4.4–3}$$

where $\sqrt{\mathcal{E}}$ denotes the symbol energy and $n_1, n_2, \ldots, n_M$ are zero-mean, mutually statistically independent Gaussian random variables with equal variance $\sigma_n^2 = \frac{1}{2}N_0$. Let us define random variables R_m, $1 \le m \le M$, as

$$R_m = \boldsymbol{r} \cdot \boldsymbol{s}_m \tag{4.4–4}$$

With this definition and from Equations 4.4–3 and 4.4–1, we have

$$\begin{aligned} R_1 &= \mathcal{E} + \sqrt{\mathcal{E}}\, n_1 \\ R_m &= \sqrt{\mathcal{E}}\, n_m, \qquad 2 \le m \le M \end{aligned} \tag{4.4–5}$$

Since we are assuming that $\boldsymbol{s}_1$ was transmitted, the detector makes a correct decision if $R_1 > R_m$ for $m = 2, 3, \ldots, M$. Therefore, the probability of a correct decision is given by

$$\begin{aligned} P_c &= \mathrm{P}\left[R_1 > R_2, R_1 > R_3, \ldots, R_1 > R_M \,|\boldsymbol{s}_1 \text{ sent}\right] \\ &= \mathrm{P}\left[\sqrt{\mathcal{E}} + n_1 > n_2, \sqrt{\mathcal{E}} + n_1 > n_3, \ldots, \sqrt{\mathcal{E}} + n_1 > n_M \middle| \boldsymbol{s}_1 \text{ sent}\right] \end{aligned} \tag{4.4–6}$$

Events $\sqrt{\mathcal{E}}+n_1 > n_2, \sqrt{\mathcal{E}}+n_1 > n_3, \ldots, \sqrt{\mathcal{E}}+n_1 > n_M$ are not independent due to the existence of the random variable n_1 in all of them. We can, however, condition on n_1 to make these events independent. Therefore, we have

$$\begin{aligned} P_c &= \int_{-\infty}^{\infty} \mathrm{P}\left[n_2 < n + \sqrt{\mathcal{E}}, n_3 < n + \sqrt{\mathcal{E}}, \ldots, n_M < n + \sqrt{\mathcal{E}} \middle| \boldsymbol{s}_1 \text{ sent}, n_1 = n\right] p_{n_1}(n)\, dn \\ &= \int_{-\infty}^{\infty} \left(\mathrm{P}\left[n_2 < n + \sqrt{\mathcal{E}} \middle| \boldsymbol{s}_1 \text{ sent}, n_1 = n\right]\right)^{M-1} p_{n_1}(n)\, dn \end{aligned} \tag{4.4–7}$$

where in the last step we have used the fact that n_m's are iid random variables for $m = 2, 3, \ldots, M$. We have

$$\mathrm{P}\left[n_2 < n + \sqrt{\mathcal{E}} \middle| \boldsymbol{s}_1 \text{ sent}, n_1 = n\right] = 1 - Q\left(\frac{n + \sqrt{\mathcal{E}}}{\sqrt{\frac{N_0}{2}}}\right) \tag{4.4–8}$$

Hence,

$$P_c = \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi N_0}} \left[1 - Q\left(\frac{n + \sqrt{\mathcal{E}}}{\sqrt{\frac{N_0}{2}}} \right) \right]^{M-1} e^{-\frac{n^2}{N_0}}\, dn \tag{4.4–9}$$

and

$$P_e = 1 - P_c = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[1 - (1 - Q(x))^{M-1} \right] e^{-\frac{\left(x - \sqrt{\frac{2\mathcal{E}}{N_0}}\right)^2}{2}}\, dx \tag{4.4–10}$$

where we have introduced a new variable $x = \frac{n+\sqrt{\mathcal{E}}}{\sqrt{\frac{N_0}{2}}}$. In general, Equation 4.4–10 cannot be made simpler, and the error probability can be found numerically for different values of the SNR.

In orthogonal signaling, due to the symmetry of the constellation, the probabilities of receiving any of the messages $m = 2, 3, \ldots, M$, when s_1 is transmitted, are equal. Therefore, for any $2 \leq m \leq M$,

$$\mathrm{P}\left[s_m \text{ received} \,|\, s_1 \text{ sent} \right] = \frac{P_e}{M-1} = \frac{P_e}{2^k - 1} \tag{4.4–11}$$

Let us assume that s_1 corresponds to a data sequence of length k with a 0 at the first component. The probability of an error at this component is the probability of detecting an s_m corresponding to a sequence with a 1 at the first component. Since there are 2^{k-1} such sequences, we have

$$P_b = 2^{k-1} \frac{P_e}{2^k - 1} = \frac{2^{k-1}}{2^k - 1} P_e \approx \frac{1}{2} P_e \tag{4.4–12}$$

where the last approximation is valid for $k \gg 1$.

The graphs of the probability of a binary digit error as a function of the SNR per bit, $\mathcal{E}_b/N_0$, are shown in Figure 4.4–1 for $M = 2, 4, 8, 16, 32$, and 64. This figure illustrates that, by increasing the number M of waveforms, one can reduce the SNR per bit required to achieve a given probability of a bit error. For example, to achieve $P_b = 10^{-5}$, the required SNR per bit is a little more than 12 dB for $M = 2$; but if M is increased to 64 signal waveforms ($k = 6$ bits per symbol), the required SNR per bit is approximately 6 dB. Thus, a savings of over 6 dB (a factor-of-4 reduction) is realized in transmitter power (or energy) required to achieve $P_b = 10^{-5}$ by increasing M from $M = 2$ to $M = 64$. This property is in direct contrast with the performance characteristics of ASK, PSK, and QAM signaling, for which increasing M increases the required power to achieve a given error probability.

Error Probability in FSK Signaling

From Equation 3.2–58 and the discussion following it, we have seen that FSK signaling becomes a special case of orthogonal signaling when the frequency separation Δf is

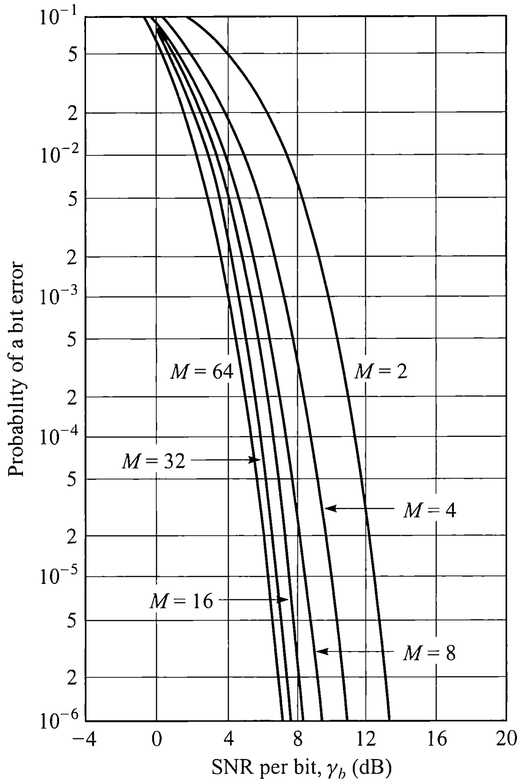


FIGURE 4.4–1
Probability of bit error for orthogonal signaling.

given by

$$\Delta f = \frac{l}{2T} \tag{4.4–13}$$

for a positive integer l. For this value of frequency separation the error probability of M-ary FSK is given by Equation 4.4–10.

Note that in the binary FSK signaling, a frequency separation that guarantees orthogonality does not minimize the error probability. In Problem 4.18 it is shown that the error probability of binary FSK is minimized when the frequency separation is of the form

$$\Delta f = \frac{0.715}{T} \tag{4.4–14}$$

A Union Bound on the Probability of Error in Orthogonal Signaling

The union bound derived in Section 4.2–3 states that

$$P_e \le \frac{M-1}{2} e^{-\frac{d^2_{\min}}{4N_0}} \tag{4.4–15}$$

In orthogonal signaling $d_{\min} = \sqrt{2\mathcal{E}}$, therefore,

$$P_e \le \frac{M-1}{2} e^{-\frac{\mathcal{E}}{2N_0}} < M e^{-\frac{\mathcal{E}}{2N_0}} \tag{4.4–16}$$

Using $M = 2^k$ and $\mathcal{E}_b = \mathcal{E}/k$, we have

$$P_e < 2^k e^{-\frac{k\mathcal{E}_b}{2N_0}} = e^{-\frac{k}{2}\left(\frac{\mathcal{E}_b}{N_0}-2\ln 2\right)} \tag{4.4–17}$$

It is clear from Equation 4.4–17 that if

$$\frac{\mathcal{E}_b}{N_0} > 2\ln 2 = 1.39 \sim 1.42 \text{ dB} \tag{4.4–18}$$

then $P_e \to \infty$ as $k \to \infty$. In other words, if the SNR per bit exceeds 1.42 dB, then *reliable communication*[†] is possible.

One can ask whether the condition SNR per bit $>$ 1.42 dB is necessary, as well as being sufficient, for reliable communication. We will see in Chapter 6 that this condition is not necessary. We will show there that a necessary and sufficient condition for reliable communication is

$$\frac{\mathcal{E}_b}{N_0} > \ln 2 = 0.693 \sim -1.6 \text{ dB} \tag{4.4–19}$$

Thus, reliable communication at SNR per bit lower than -1.6 dB is impossible. The reason that Equation 4.4–17 does not result in this tighter bound is that the union bound is not tight enough at low SNRs. To obtain the -1.6 dB bound, more sophisticated bounding techniques are required. By using these bounding techniques it can be shown that

$$P_e \leq \begin{cases} e^{-\frac{k}{2}\left(\frac{\mathcal{E}_b}{N_0}-2\ln 2\right)} & \frac{\mathcal{E}_b}{N_0} > 4\ln 2 \\ 2e^{-k\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}-\sqrt{\ln 2}\right)^2} & \ln 2 \leq \frac{\mathcal{E}_b}{N_0} \leq 4\ln 2 \end{cases} \tag{4.4–20}$$

The minimum value of SNR per bit needed for reliable communication, i.e., -1.6 dB, is called the *Shannon limit*. We will discuss this topic and the notion of channel capacity in greater detail in Chapter 6.

4.4–2 Optimal Detection and Error Probability for Biorthogonal Signaling

As indicated in Section 3.2–4, a set of $M = 2^k$ biorthogonal signals is constructed from $\frac{1}{2}M$ orthogonal signals by including the negatives of the orthogonal signals. Thus, we achieve a reduction in the complexity of the demodulator for the biorthogonal signals relative to that for orthogonal signals, since the former is implemented with $\frac{1}{2}M$ cross-correlators or matched filters, whereas the latter requires M matched filters, or cross-correlators. In biorthogonal signaling $N = \frac{1}{2}M$, and the vector representation for signals are given by

---

[†] We say reliable communication is possible if we can make the error probability as small as desired

$$\begin{aligned} \boldsymbol{s}_1 &= -\boldsymbol{s}_{N+1} = (\sqrt{\mathcal{E}}, 0, \ldots, 0) \\ \boldsymbol{s}_2 &= -\boldsymbol{s}_{N+2} = (0, \sqrt{\mathcal{E}}, \ldots, 0) \\ \vdots &= \quad \vdots \quad = \quad \vdots \\ \boldsymbol{s}_N &= -\boldsymbol{s}_{2N} \; = (0, \ldots, 0, \sqrt{\mathcal{E}}) \end{aligned} \tag{4.4–21}$$

To evaluate the probability of error for the optimum detector, let us assume that the signal $s_1(t)$ corresponding to the vector $\boldsymbol{s}_1 = (\sqrt{\mathcal{E}}, 0, \ldots, 0)$ was transmitted. Then the received signal vector is

$$\boldsymbol{r} = (\sqrt{\mathcal{E}} + n_1, n_2, \ldots, n_N) \tag{4.4–22}$$

where the $\{n_m\}$ are zero-mean, mutually statistically independent and identically distributed Gaussian random variables with variance $\sigma_n^2 = \frac{1}{2}N_0$. Since all signals are equiprobable and have equal energy, the optimum detector decides in favor of the signal corresponding to the largest in magnitude of the cross-correlators

$$C(\boldsymbol{r}, \boldsymbol{s}_m) = \boldsymbol{r} \cdot \boldsymbol{s}_m, \qquad 1 \le m \le \frac{1}{2}M \tag{4.4–23}$$

while the sign of this largest term is used to decide whether $s_m(t)$ or $-s_m(t)$ was transmitted. According to this decision rule, the probability of a correct decision is equal to the probability that $r_1 = \sqrt{\mathcal{E}} + n_1 > 0$ and r_1 exceeds $|r_m| = |n_m|$ for $m = 2, 3, \ldots, \frac{1}{2}M$. But

$$\begin{aligned} \mathrm{P}\left[|n_m| < r_1 \,|r_1 > 0\right] &= \frac{1}{\sqrt{\pi N_0}} \int_{-r_1}^{r_1} e^{-x^2/N_0}\, dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\frac{r_1}{\sqrt{N_0/2}}}^{\frac{r_1}{\sqrt{N_0/2}}} e^{-\frac{x^2}{2}}\, dx \end{aligned} \tag{4.4–24}$$

Then the probability of a correct decision is

$$P_c = \int_0^\infty \left(\frac{1}{\sqrt{2\pi}} \int_{-\frac{r_1}{\sqrt{N_0/2}}}^{\frac{r_1}{\sqrt{N_0/2}}} e^{-\frac{x^2}{2}}\, dx \right)^{M/2-1} p(r_1)\, dr_1 \tag{4.4–25}$$

from which, upon substitution for $p(r_1)$, we obtain

$$P_c = \frac{1}{\sqrt{2\pi}} \int_{-\sqrt{2\mathcal{E}/N_0}}^{\infty} \left(\frac{1}{\sqrt{2\pi}} \int_{-(v+\sqrt{2\mathcal{E}/N_0})}^{v+\sqrt{2\mathcal{E}/N_0}} e^{-\frac{x^2}{2}}\, dx \right)^{M/2-1} e^{-\frac{v^2}{2}}\, dv \tag{4.4–26}$$

where we have used the PDF of r_1 as a Gaussian random variable with mean equal to $\sqrt{\mathcal{E}}$ and variance $\frac{1}{2}N_0$. Finally, the probability of a symbol error $P_e = 1 - P_c$. P_c, and hence, P_e may be evaluated numerically for different values of M from Equation 4.4–26. The graph shown in Figure 4.4–2 illustrates P_e as a function of $\mathcal{E}_b/N_0$, where $\mathcal{E} = k\mathcal{E}_b$, for $M = 2, 4, 8, 16$, and 32. We observe that this graph is similar to that for orthogonal signals (see Figure 4.4–1). However, in this case, the probability of error for $M = 4$ is greater than that for $M = 2$. This is due to the fact that we have plotted the symbol

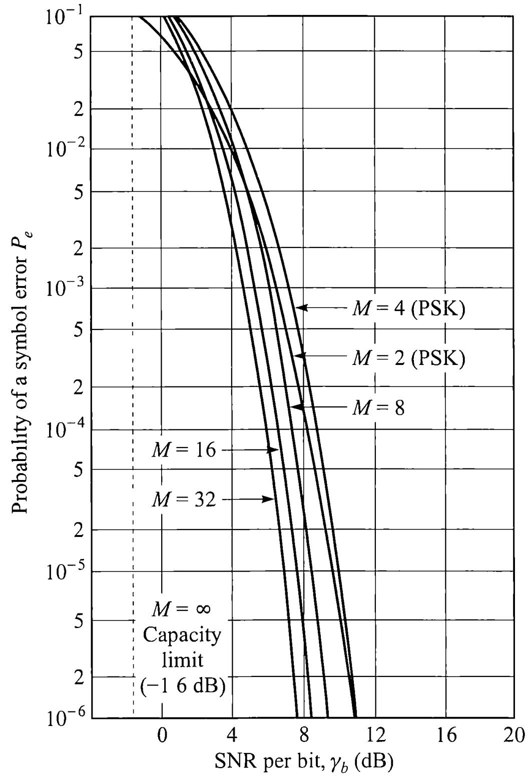


FIGURE 4.4–2
Probability of symbol error for biorthogonal signals.

error probability P_e in Figure 4.4–2. If we plotted the equivalent bit error probability, we should find that the graphs for $M = 2$ and $M = 4$ coincide. As in the case of orthogonal signals, as $M \to \infty$ (or $k \to \infty$), the minimum required $\mathcal{E}_b/N_0$ to achieve an arbitrarily small probability of error is -1.6 dB, the Shannon limit.

4.4–3 Optimal Detection and Error Probability for Simplex Signaling

As we have seen in Section 3.2–4, simplex signals are obtained from a set of orthogonal signals by shifting each signal by the average of the orthogonal signals. Since the signals of an orthogonal signal are simply shifted by a constant vector to obtain the simplex signals, the geometry of the simplex signal, i.e., the distance between signals and the angle between lines joining signals, is exactly the same as that of the original orthogonal signals. Therefore, the error probability of a set of simplex signals is given by the same expression as the expression derived for orthogonal signals. However, since simplex signals have a lower energy, as indicated by Equation 3.2–65 the energy in the expression for error probability should be scaled accordingly. Therefore the expression for the error probability in simplex signaling becomes

$$P_e = 1 - P_c = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \left[1 - (1 - Q(x))^{M-1}\right] e^{-\frac{\left(x - \sqrt{\frac{M}{M-1}\frac{2\mathcal{E}}{N_0}}\right)^2}{2}} dx \qquad (4.4\text{–}27)$$

This indicates a relative gain of $10 \log \frac{M}{M-1}$ over orthogonal signaling. For $M = 2$, this gain becomes 3 dB; for $M = 10$ it reduces to 0.46 dB; and as M becomes larger, it becomes negligible and the performance of orthogonal and simplex signals becomes similar. Obviously, for simplex signals, similar to orthogonal and biorthogonal signals, the error probability decreases as M increases.

4.5
OPTIMAL DETECTION IN PRESENCE OF UNCERTAINTY: NONCOHERENT DETECTION

In the detection schemes we have studied so far, we made the implicit assumption that the signals $\{s_m(t), 1 \le m \le M\}$ are available at the receiver. This assumption was in the form of either the availability of the signals themselves or the availability of an orthonormal basis $\{\phi_j(t), 1 \le j \le N\}$. Although in many communication systems this assumption is valid, there are many cases in which we cannot make such an assumption.

One of the cases in which such an assumption is invalid occurs when transmission over the channel introduces random changes to the signal as either a random attenuation or a random phase shift. These situations will be studied in detail in Chapter 13. Another situation that results in imperfect knowledge of the signals at the receiver arises when the transmitter and the receiver are not perfectly synchronized. In this case, although the receiver knows the general shape of $\{s_m(t)\}$, due to imperfect synchronization with the transmitter, it can use only signals in the form of $\{s_m(t - t_d)\}$, where t_d represents the time slip between the transmitter and the receiver clocks. This time slip can be modeled as a random variable.

To study the effect of random parameters of this type on the optimal receiver design and performance, we consider the transmission of a set of signals over the AWGN channel with some random parameter denoted by the random vector $\boldsymbol{\theta}$. We assume that signals $\{s_m(t), 1 \le m \le M\}$ are transmitted, and the received signal $r(t)$ can be written as

$$r(t) = s_m(t; \boldsymbol{\theta}) + n(t) \tag{4.5–1}$$

where $\boldsymbol{\theta}$ is in general a vector-valued random variable. By the Karhunen-Loeve expansion theorem discussed in Section 2.8–2, we can find an orthonormal basis for expansion of the random process $s_m(t; \boldsymbol{\theta})$ and by Example 2.8–1, the same orthonormal basis can be used for expansion of the white Gaussian noise process $n(t)$. By using this basis, the waveform channel given in Equation 4.5–1 becomes equivalent to the vector channel

$$\boldsymbol{r} = \boldsymbol{s}_{m,\theta} + \boldsymbol{n} \tag{4.5–2}$$

for which the optimal detection rule is given by

$$\begin{aligned}
\hat{m} &= \underset{1 \le m \le M}{\arg\max}\ P_m p(\boldsymbol{r}|m) \\
&= \underset{1 \le m \le M}{\arg\max}\ P_m \int p(\boldsymbol{r}|m, \boldsymbol{\theta}) p(\boldsymbol{\theta})\, d\boldsymbol{\theta} \\
&= \underset{1 \le m \le M}{\arg\max}\ P_m \int p_{\boldsymbol{n}}(\boldsymbol{r} - \boldsymbol{s}_{m,\boldsymbol{\theta}}) p(\boldsymbol{\theta})\, d\boldsymbol{\theta}
\end{aligned} \tag{4.5–3}$$

Equation 4.5–3 represents the optimal decision rule and the resulting decision regions. The minimum error probability, when the optimal detection rule of Equation 4.5–3 is employed, is given by

$$\begin{aligned} P_e &= \sum_{m=1}^{M} P_m \int_{D_m^c} \left(\int p(\boldsymbol{r} \mid m, \boldsymbol{\theta}) p(\boldsymbol{\theta})\, d\boldsymbol{\theta} \right) d\boldsymbol{r} \\ &= \sum_{m=1}^{M} P_m \sum_{\substack{m'=1 \\ m' \neq m}}^{M} \int_{D_{m'}} \left(\int p_{\boldsymbol{n}}(\boldsymbol{r} - \boldsymbol{s}_{m,\boldsymbol{\theta}}) p(\boldsymbol{\theta})\, d\boldsymbol{\theta} \right) d\boldsymbol{r} \end{aligned} \tag{4.5–4}$$

Equations 4.5–3 and 4.5–4 are quite general and can be used for all types of uncertainties in channel parameters.

EXAMPLE 4.5–1. A binary antipodal signaling system with equiprobable signals $s_1(t) = s(t)$ and $s_2(t) = -s(t)$ is used on an AWGN channel with noise power spectral density of $\frac{N_0}{2}$. The channel introduces a random gain of A which can take only non-negative values. In other words the channel does not invert the polarity of the signal. This channel can be modeled as

$$r(t) = A\, s_m(t) + n(t) \tag{4.5–5}$$

where A is a random gain with PDF $p(A)$ and $p(A) = 0$ for $A < 0$. Using Equation 4.5–3, and noting that $p(\boldsymbol{r} \mid m, A) = p_n(r - As_m)$, D_1, the optimal decision region for $s_1(t)$ is given by

$$D_1 = \left\{ r : \int_0^{\infty} e^{-\frac{(r - A\sqrt{\mathcal{E}_b})^2}{N_0}} p(A)\, dA > \int_0^{\infty} e^{-\frac{(r + A\sqrt{\mathcal{E}_b})^2}{N_0}} p(A)\, dA \right\} \tag{4.5–6}$$

which simplifies to

$$D_1 = \left\{ r : \int_0^{\infty} e^{-\frac{A^2 \mathcal{E}_b}{N_0}} \left(e^{\frac{2rA\sqrt{\mathcal{E}_b}}{N_0}} - e^{-\frac{2rA\sqrt{\mathcal{E}_b}}{N_0}} \right) p(A)\, dA > 0 \right\} \tag{4.5–7}$$

Since A takes only positive values, the expression inside the parentheses is positive if and only if $r > 0$. Therefore,

$$D_1 = \{ r : r > 0 \} \tag{4.5–8}$$

To compute the error probability, we have

$$\begin{aligned} P_b &= \int_0^{\infty} \left(\int_0^{\infty} \frac{1}{\sqrt{\pi N_0}} e^{-\frac{(r + A\sqrt{\mathcal{E}_b})^2}{N_0}} dr \right) p(A)\, dA \\ &= \int_0^{\infty} Q\left(A \sqrt{\frac{2\mathcal{E}_b}{N_0}} \right) p(A)\, dA \\ &= \mathrm{E}\left[Q\left(A \sqrt{\frac{2\mathcal{E}_b}{N_0}} \right) \right] \end{aligned} \tag{4.5–9}$$

where the expectation is taken with respect to A. For instance, if A takes values $\frac{1}{2}$ and 1 with equal probability, then

$$P_b = \frac{1}{2}Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) + \frac{1}{2}Q\left(\sqrt{\frac{\mathcal{E}_b}{2N_0}}\right)$$

It is important to note that in this case the average received energy per bit is $\mathcal{E}_{\text{bavg}} = \frac{1}{2}\mathcal{E}_b + \frac{1}{2}(\frac{1}{4}\mathcal{E}_b) = \frac{5}{8}\mathcal{E}_b$. In Problem 6.29 we show that $P_b \geq Q\left(\sqrt{\frac{2\mathcal{E}_{\text{bavg}}}{N_0}}\right)$.

4.5–1 Noncoherent Detection of Carrier Modulated Signals

For carrier modulated signals, $\{s_m(t), 1 \leq m \leq M\}$ are bandpass signals with lowpass equivalents $\{s_{ml}(t), 1 \leq m \leq M\}$ where

$$s_m(t) = \text{Re}\left[s_{ml}(t)e^{j2\pi f_c t}\right] \tag{4.5–10}$$

The AWGN channel model in general is given by

$$r(t) = s_m(t - t_d) + n(t) \tag{4.5–11}$$

where t_d indicates the random time asynchronism between the clocks of the transmitter and the receiver. It is clearly seen that the received random process $r(t)$ is a function of three random phenomena, the message m, which is selected with probability P_m, the random variable t_d, and finally the random process $n(t)$.

From Equations 4.5–10 and 4.5–11 we have

$$\begin{aligned} r(t) &= \text{Re}\left[s_{ml}(t - t_d)e^{j2\pi f_c(t-t_d)}\right] + n(t) \\ &= \text{Re}\left[s_{ml}(t - t_d)e^{-j2\pi f_c t_d}e^{j2\pi f_c t}\right] + n(t) \end{aligned} \tag{4.5–12}$$

Therefore, the lowpass equivalent of $s_m(t - t_d)$ is equal to $s_{ml}(t - t_d)e^{-j2\pi f_c t_d}$. In practice $t_d \ll T_s$, where T_s is the symbol duration. This means that the effect of a time shift of size t_d on $s_{ml}(t)$ is negligible. However, the term $e^{-j2\pi f_c t_d}$ can introduce a large phase shift $\phi = -2\pi f_c t_d$ because even small values of t_d are multiplied by large carrier frequency f_c, resulting in noticeable phase shifts. Since t_d is random and even small values of t_d can cause large phase shifts that are folded modulo 2π, we can model ϕ as a random variable uniformly distributed between 0 and 2π. This model of the channel and detection of signals under this assumption is called *noncoherent detection*.

From this discussion we conclude that in the noncoherent case

$$\text{Re}\left[r_l(t)e^{j2\pi f_c t}\right] = \text{Re}\left[\left(e^{j\phi}s_{ml}(t) + n_l(t)\right)e^{j2\pi f_c t}\right] \tag{4.5–13}$$

or, in the baseband

$$r_l(t) = e^{j\phi}s_{ml}(t) + n_l(t) \tag{4.5–14}$$

Note that by the discussion following Equation 2.9–14, the lowpass noise process $n_l(t)$ is circular and its statistics are independent of any rotation; hence we can ignore the effect of phase rotation on the noise component. For the phase coherent case where the receiver knows ϕ, it can compensate for it, and the lowpass equivalent channel will have the familiar form of

$$r_l(t) = s_{ml}(t) + n_l(t) \tag{4.5–15}$$

In the noncoherent case, the vector equivalent of Equation 4.5–15 is given by

$$\boldsymbol{r}_l = e^{j\phi}\boldsymbol{s}_{ml} + \boldsymbol{n}_l \tag{4.5–16}$$

To design the optimal detector for the baseband vector channel of Equation 4.5–16, we use the general formulation of the optimal detector given in Equation 4.5–3 as

$$\hat{m} = \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}\int_0^{2\pi} p_{\boldsymbol{n}_l}(\boldsymbol{r}_l - e^{j\phi}\boldsymbol{s}_{ml})\,d\phi \tag{4.5–17}$$

From Example 2.9–1 it is seen that $n_l(t)$ is a complex baseband random process with power spectral density of $2N_0$ in the $[-W, W]$ frequency band. The projections of this process on an orthonormal basis will have complex iid zero-mean Gaussian components with variance $2N_0$ (variance N_0 per real and imaginary components). Therefore we can write

$$\hat{m} = \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}\frac{1}{(4\pi N_0)^N}\int_0^{2\pi} e^{-\frac{\|\boldsymbol{r}_l - e^{j\phi}\boldsymbol{s}_{ml}\|^2}{4N_0}}\,d\phi \tag{4.5–18}$$

Expanding the exponent, dropping terms that do not depend on m, and noting that $\|\boldsymbol{s}_{ml}\|^2 = 2\mathcal{E}_m$, we obtain

$$\begin{aligned}
\hat{m} &= \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}e^{-\frac{\mathcal{E}_m}{2N_0}}\int_0^{2\pi} e^{\frac{1}{2N_0}\mathrm{Re}[\boldsymbol{r}_l\, e^{j\phi}\boldsymbol{s}_{ml}]}\,d\phi\\
&= \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}e^{-\frac{\mathcal{E}_m}{2N_0}}\int_0^{2\pi} e^{\frac{1}{2N_0}\mathrm{Re}[(\boldsymbol{r}_l\, \boldsymbol{s}_{ml})e^{-j\phi}]}\,d\phi\\
&= \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}e^{-\frac{\mathcal{E}_m}{2N_0}}\int_0^{2\pi} e^{\frac{1}{2N_0}\mathrm{Re}[|\boldsymbol{r}_l\, \boldsymbol{s}_{ml}|e^{-j(\phi-\theta)}]}\,d\phi\\
&= \underset{1\le m\le M}{\arg\max}\ \frac{P_m}{2\pi}e^{-\frac{\mathcal{E}_m}{2N_0}}\int_0^{2\pi} e^{\frac{1}{2N_0}|\boldsymbol{r}_l\, \boldsymbol{s}_{ml}|\cos(\phi-\theta)}\,d\phi
\end{aligned} \tag{4.5–19}$$

where θ denotes the phase of $\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}$. Note that the integrand in Equation 4.5–19 is a periodic function of ϕ with period 2π, and we are integrating over a complete period; therefore θ has no effect on the result. Using the relation

$$I_0(x) = \frac{1}{2\pi}\int_0^{2\pi} e^{x\cos\phi}\,d\phi \tag{4.5–20}$$

where $I_0(x)$ is the modified Bessel function of the first kind and order zero, we obtain

$$\hat{m} = \underset{1\le m\le M}{\arg\max}\ P_m\, e^{-\frac{\mathcal{E}_m}{2N_0}} I_0\left(\frac{|\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}|}{2N_0}\right) \tag{4.5–21}$$

In general, the decision rule given in Equation 4.5–21 cannot be made simpler. However, in the case of equiprobable and equal-energy signals, the terms P_m and $\mathcal{E}_m$ can be ignored, and the optimal detection rule becomes

$$\hat{m} = \arg\max_{1\le m\le M} I_0\left(\frac{|\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}|}{2N_0}\right) \tag{4.5–22}$$

Since for $x > 0$, $I_0(x)$ is an increasing function of x, the decision rule in this case reduces to

$$\hat{m} = \arg\max_{1\le m\le M} |\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}| \tag{4.5–23}$$

From Equation 4.5–23 it is clear that an optimal noncoherent detector first demodulates the received signal, using its nonsynchronized local oscillator, to obtain $r_l(t)$, the lowpass equivalent of the received signal. It then correlates $r_l(t)$ with all $s_{ml}(t)$'s and chooses the one that has the maximum absolute value, or envelope. This detector is called an *envelope detector*. Note that Equation 4.5–23 can also be written as

$$\hat{m} = \arg\max_{1\le m\le M} \left|\int_{-\infty}^{\infty} r_l(t)s_{ml}^*(t)\,dt\right| \tag{4.5–24}$$

The block diagram of an envelope detector is shown in Figure 4.5–1. Detailed block diagrams for the demodulator and the complex matched filters shown in this figure are given in Figures 4.3–9 and 4.3–11, respectively.

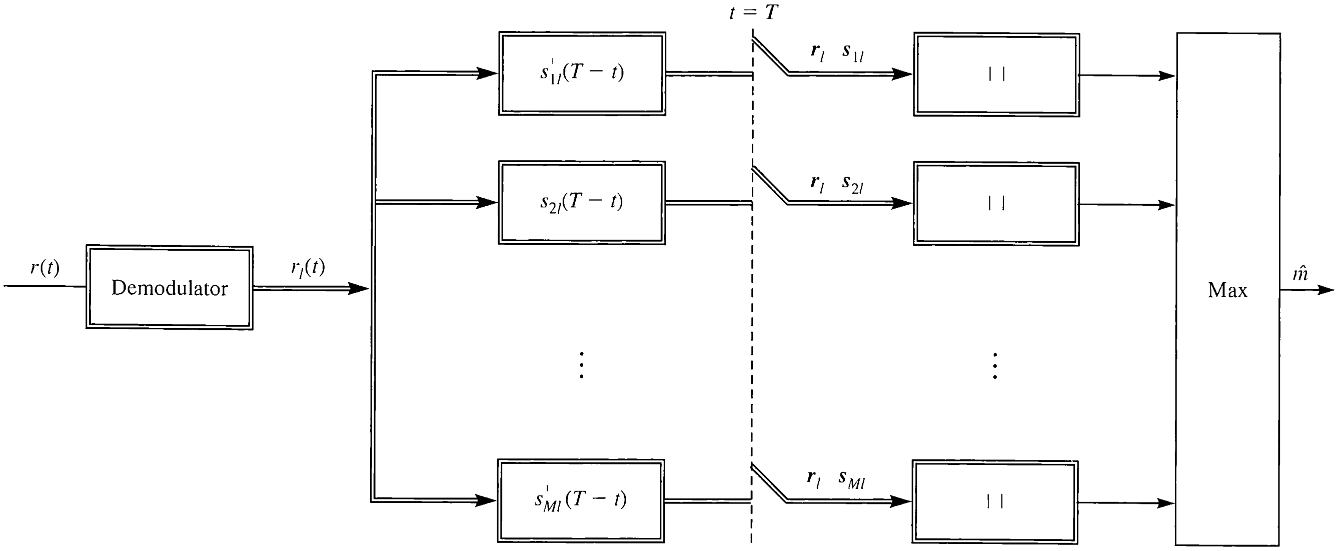


FIGURE 4.5–1
Block diagram of an envelope detector.

4.5–2 Optimal Noncoherent Detection of FSK Modulated Signals

For equiprobable FSK signaling, the signals have equal energy and the optimal detection rule is given by Equation 4.5–23. Assuming that frequency separation between signals is Δf, the FSK signals have the general form

$$\begin{aligned} s_m(t) &= g(t)\cos(2\pi f_c t + 2\pi(m-1)\Delta f t) \\ &= \operatorname{Re}\left[g(t)e^{j2\pi(m-1)\Delta f t}e^{j2\pi f_c t}\right], \qquad 1 \le m \le M \end{aligned} \tag{4.5–25}$$

Hence,

$$s_{ml}(t) = g(t)e^{j2\pi(m-1)\Delta f t} \tag{4.5–26}$$

where $g(t)$ is a rectangular pulse of duration T_s and $\mathcal{E}_g = 2\mathcal{E}_s$, where $\mathcal{E}_s$ denotes the energy per transmitted symbol. At the receiver, the optimal noncoherent detector correlates $r_l(t)$ with $s_{m'l}(t)$ for all $1 \le m' \le M$. Assuming $s_m(t)$ is transmitted, from Equation 4.5–24 we have

$$\begin{aligned} \left|\int_{-\infty}^{\infty} r_l(t)s_{m'l}^*(t)\,dt\right| &= \left|\int_{-\infty}^{\infty} (s_{ml}(t) + n_l(t))\,s_{m'l}^*(t)\,dt\right| \\ &= \left|\int_{-\infty}^{\infty} s_{ml}(t)s_{m'l}^*(t)\,dt + \int_{-\infty}^{\infty} n_l(t)s_{m'l}^*(t)\,dt\right| \end{aligned} \tag{4.5–27}$$

But

$$\begin{aligned} \int_{-\infty}^{\infty} s_{ml}(t)s_{m'l}^*(t)\,dt &= \frac{2\mathcal{E}_s}{T_s}\int_0^{T_s} e^{j2\pi(m-1)\Delta f t}e^{-j2\pi(m'-1)\Delta f t} \\ &= \frac{2\mathcal{E}_s}{T_s}\int_0^{T_s} e^{j2\pi(m-m')\Delta f t}\,dt \\ &= \frac{2\mathcal{E}_s}{T_s}\frac{1}{j2\pi(m-m')\Delta f}\left[e^{j2\pi(m-m')\Delta f T_s} - 1\right] \\ &= 2\mathcal{E}_s\, e^{j\pi(m-m')\Delta f T_s}\operatorname{sinc}\left[(m-m')\Delta f T_s\right] \end{aligned} \tag{4.5–28}$$

From Equation 4.5–28 we see that if and only if $\Delta f = \frac{k}{T_s}$ for some integer k, then $\langle s_{ml}(t), s_{m'l}(t)\rangle = 0$ for all $m' \ne m$. This is the condition of orthogonality for FSK signals under noncoherent detection. For coherent detection, however, the detector uses Equation 4.3–41, and for orthogonality we must have $\operatorname{Re}\left[\langle s_{ml}(t), s_{m'l}(t)\rangle\right] = 0$. But from Equation 3.2–58

$$\begin{aligned} \operatorname{Re}\left[\int_{-\infty}^{\infty} s_{ml}(t)s_{m'l}^*(t)\,dt\right] &= 2\mathcal{E}_s\cos\left(\pi(m-m')\Delta f T_s\right)\operatorname{sinc}\left[(m-m')\Delta f T_s\right] \\ &= 2\mathcal{E}_s\operatorname{sinc}\left[2(m-m')\Delta f T_s\right] \end{aligned} \tag{4.5–29}$$

Obviously, the condition for orthogonality in this case is $\Delta f = \frac{k}{2T_s}$. It is clear from the above discussion that orthogonality under noncoherent detection guarantees orthogonality under coherent detection, but not vice versa.

The optimal noncoherent detection rule for FSK signaling follows the general rule for noncoherent detection of equiprobable and equal-energy signals and is implemented using an envelope or a square-law detector.

4.5–3 Error Probability of Orthogonal Signaling with Noncoherent Detection

Let us assume M equiprobable, equal-energy, carrier modulated orthogonal signals are transmitted over an AWGN channel. These signals are noncoherently demodulated at the receiver and and then optimally detected. For instance, in coherent detection of orthogonal FSK signals we encounter a situation like this. The lowpass equivalent of the signals can be written as M N-dimensional vectors ($N = M$)

$$\begin{aligned} s_{1l} &= \left(\sqrt{2\mathcal{E}_s}, 0, 0, \ldots, 0\right) \\ s_{2l} &= \left(0, \sqrt{2\mathcal{E}_s}, 0, \ldots, 0\right) \\ \vdots \; &= \quad \vdots \\ s_{Ml} &= \left(0, 0, \ldots, 0, \sqrt{2\mathcal{E}_s}\right) \end{aligned} \tag{4.5–30}$$

Because of the symmetry of the constellation, without loss of generality we can assume that s_{1l} is transmitted. Therefore, the received vector will be

$$\boldsymbol{r}_l = e^{j\phi} \boldsymbol{s}_{1l} + \boldsymbol{n}_l \tag{4.5–31}$$

where $\boldsymbol{n}_l$ is a complex circular zero-mean Gaussian random vector with variance of each complex component equal to $2N_0$ (this follows from the result of Example 2.9–1). The optimal receiver computes and compares $|\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}|$, for all $1 \le m \le M$. This results in

$$\begin{aligned} |\boldsymbol{r}_l \cdot \boldsymbol{s}_{1l}| &= |2\mathcal{E}_s e^{j\phi} + \boldsymbol{n}_l \cdot \boldsymbol{s}_{1l}| \\ |\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}| &= |\boldsymbol{n}_l \cdot \boldsymbol{s}_{ml}|, \qquad 2 \le m \le M \end{aligned} \tag{4.5–32}$$

For $1 \le m \le M$, $\boldsymbol{n}_l \cdot \boldsymbol{s}_{ml}$ is a circular zero-mean complex Gaussian random variable with variance $4\mathcal{E}_s N_0$ ($2\mathcal{E}_s N_0$ per real and imaginary parts). From Equation 4.5–32 it is seen that

$$\begin{aligned} \operatorname{Re}[\boldsymbol{r}_l \cdot \boldsymbol{s}_{1l}] &\sim \mathcal{N}(2\mathcal{E}_s \cos\phi, 2\mathcal{E}_s N_0) \\ \operatorname{Im}[\boldsymbol{r}_l \cdot \boldsymbol{s}_{1l}] &\sim \mathcal{N}(2\mathcal{E}_s \sin\phi, 2\mathcal{E}_s N_0) \\ \operatorname{Re}[\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}] &\sim \mathcal{N}(0, 2\mathcal{E}_s N_0), \qquad 2 \le m \le M \\ \operatorname{Im}[\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}] &\sim \mathcal{N}(0, 2\mathcal{E}_s N_0), \qquad 2 \le m \le M \end{aligned} \tag{4.5–33}$$

From the definition of Rayleigh and Ricean random variables given Chapter 2 in Equations 2.3–42 and 2.3–55, we conclude that random variables R_m, $1 \le m \le M$, defined as

$$R_m = |\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}|, \qquad 1 \le m \le M \tag{4.5–34}$$

are independent random variables, R_1 has a Ricean distribution with parameters $s = 2\mathcal{E}_s$ and $\sigma^2 = 2\mathcal{E}_s N_0$, and R_m, $2 \le m \le M$, are Rayleigh random variables[†] with parameter $\sigma^2 = 2\mathcal{E}_s N_0$. In other words,

$$p_{R_1}(r_1) = \begin{cases} \dfrac{r_1}{\sigma^2} I_0\left(\frac{sr_1}{\sigma^2}\right) e^{-\frac{r_1^2+s^2}{2\sigma^2}} & r_1 > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.5–35}$$

and

$$p_{R_m}(r_m) = \begin{cases} \frac{r_m}{\sigma^2} e^{-\frac{r_m^2}{2\sigma^2}} & r_m > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.5–36}$$

for $2 \le m \le M$. Since by assumption s_{1l} is transmitted, a correct decision is made at the receiver if $R_1 > R_m$ for $2 \le m \le M$. Although random variables R_m for $1 \le m \le M$ are statistically independent, the events $R_1 > R_2$, $R_1 > R_3, \ldots, R_1 > R_M$ are not independent due to the existence of the common R_1. To make them independent, we need to condition on $R_1 = r_1$ and then average over all values of r_1. Therefore,

$$\begin{aligned} P_c &= \mathrm{P}[R_2 < R_1, R_3 < R_1, \ldots, R_M < R_1] \\ &= \int_0^\infty \mathrm{P}[R_2 < r_1, R_3 < r_1, \ldots, R_M < r_1 \,|R_1 = r_1] p_{R_1}(r_1)\, dr_1 \\ &= \int_0^\infty (\mathrm{P}[R_2 < r_1])^{M-1}\, p_{R_1}(r_1)\, dr_1 \end{aligned} \tag{4.5–37}$$

But

$$\begin{aligned} \mathrm{P}[R_2 < r_1] &= \int_0^{r_1} p_{R_2}(r_2)\, dr_2 \\ &= 1 - e^{-\frac{r_1^2}{2\sigma^2}} \end{aligned} \tag{4.5–38}$$

Using the binomial expansion, we have

$$\left(1 - e^{-\frac{r_1^2}{2\sigma^2}}\right)^{M-1} = \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} e^{-\frac{nr_1^2}{2\sigma^2}} \tag{4.5–39}$$

Substituting into Equation 4.5–37, we obtain

$$\begin{aligned} P_c &= \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} \int_0^\infty e^{-\frac{nr_1^2}{2\sigma^2}} \frac{r_1}{\sigma^2} I_0\left(\frac{sr_1}{\sigma^2}\right) e^{-\frac{r_1^2+s^2}{2\sigma^2}}\, dr_1 \\ &= \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} \int_0^\infty \frac{r_1}{\sigma^2} I_0\left(\frac{sr_1}{\sigma^2}\right) e^{-\frac{(n+1)r_1^2+s^2}{2\sigma^2}}\, dr_1 \\ &= \sum_{n=0}^{M-1} (-1)^n \binom{M-1}{n} e^{-\frac{ns^2}{2(n+1)\sigma^2}} \int_0^\infty \frac{r_1}{\sigma^2} I_0\left(\frac{sr_1}{\sigma^2}\right) e^{-\frac{(n+1)r_1^2+\frac{s^2}{n+1}}{2\sigma^2}}\, dr_1 \end{aligned} \tag{4.5–40}$$

[†] To be more precise, we have to note that ϕ is itself a uniform random variable, therefore to obtain the PDF of R_m, we need to first condition on ϕ and then average with respect to the uniform PDF This, however, does not change the final result stated above

By introducing a change of variables

$$\begin{aligned} s' &= \frac{s}{\sqrt{n+1}} \\ r' &= r_1\sqrt{n+1} \end{aligned} \tag{4.5–41}$$

the integral in Equation 4.5–40 becomes

$$\begin{aligned} \int_0^\infty \frac{r_1}{\sigma^2} I_0\left(\frac{sr_1}{\sigma^2}\right) e^{-\frac{(n+1)r_1^2+\frac{s^2}{n+1}}{2\sigma^2}}\, dr_1 &= \frac{1}{n+1}\int_0^\infty \frac{r'}{\sigma^2} I_0\left(\frac{r's'}{\sigma^2}\right) e^{-\frac{s'^2+r'^2}{2\sigma^2}}\, dr' \\ &= \frac{1}{n+1} \end{aligned} \tag{4.5–42}$$

where in the last step we used the fact that the area under a Ricean PDF is equal to 1. Substituting Equation 4.5–42 into Equation 4.5–40 and noting that $\frac{s^2}{2\sigma^2} = \frac{4\mathcal{E}_s^2}{4\mathcal{E}_s N_0} = \frac{\mathcal{E}_s}{N_0}$, we obtain

$$P_c = \sum_{n=0}^{M-1} \frac{(-1)^n}{n+1}\binom{M-1}{n} e^{-\frac{n}{n+1}\frac{\mathcal{E}_s}{N_0}} \tag{4.5–43}$$

Then the probability of a symbol error becomes

$$P_e = \sum_{n=1}^{M-1} \frac{(-1)^{n+1}}{n+1}\binom{M-1}{n} e^{-\frac{n\log_2 M}{n+1}\frac{\mathcal{E}_b}{N_0}} \tag{4.5–44}$$

For binary orthogonal signaling, including binary orthogonal FSK with noncoherent detection, Equation 4.5–44 simplifies to

$$P_b = \frac{1}{2} e^{-\frac{\mathcal{E}_b}{2N_0}} \tag{4.5–45}$$

Comparing this result with coherent detection of binary orthogonal signals for which the error probability is given by

$$P_b = Q\left(\sqrt{\frac{\mathcal{E}_b}{N_0}}\right) \tag{4.5–46}$$

and using the inequality $Q(x) \le \frac{1}{2}e^{-x^2/2}$, we conclude that $P_{b\text{noncoh}} \ge P_{b\text{coh}}$, as expected. For error probabilities less than 10^{-4}, the difference between the performance of coherent and noncoherent detection of binary orthogonal is less than 0.8 dB.

For $M > 2$, we may compute the probability of a bit error by making use of the relationship

$$P_b = \frac{2^{k-1}}{2^k - 1} P_e \tag{4.5–47}$$

which was established in Section 4.4–1. Figure 4.5–2 shows the bit error probability as a function of the SNR per bit γ_b for $M = 2, 4, 8, 16$, and 32. Just as in the case of coherent detection of M-ary orthogonal signals (see Figure 4.4–1), we observe that for any given bit error probability, the SNR per bit decreases as M increases. It will be shown in Chapter 6 that, in the limit as $M \to \infty$ (or $k = \log_2 M \to \infty$), the

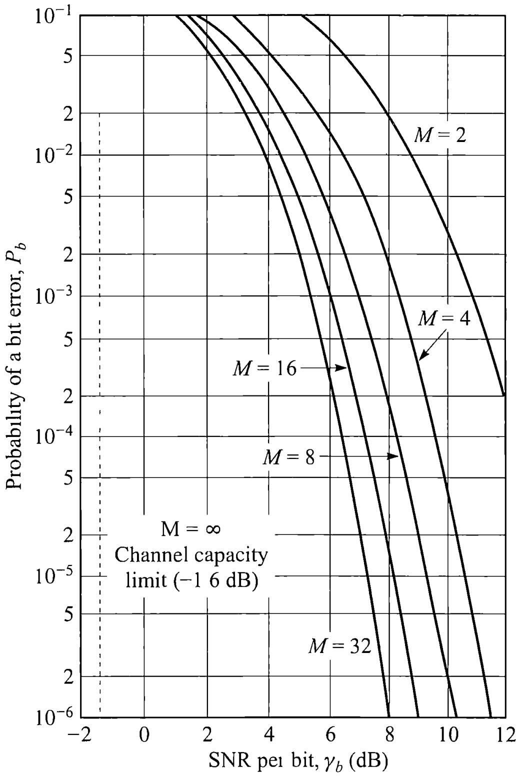


FIGURE 4.5–2
Probability of a bit error for noncoherent detection of orthogonal signals.

probability of a bit error P_b can be made arbitrarily small provided that the SNR per bit is greater than the Shannon limit of -1.6 dB. The cost for increasing M is the bandwidth required to transmit the signals. For M-ary FSK, the frequency separation between adjacent frequencies is $\Delta f = 1/T_s$ for signal orthogonality. The bandwidth required for the M signals is $W = M\,\Delta f = M/T_s$.

4.5–4 Probability of Error for Envelope Detection of Correlated Binary Signals

In this section, we consider the performance of the envelope detector for binary, equiprobable, and equal-energy correlated signals. When the two signals are correlated, we have

$$\boldsymbol{s}_{ml} \cdot \boldsymbol{s}_{m'l} = \begin{cases} 2\mathcal{E}_s & m = m' \\ 2\mathcal{E}_s \rho & m \neq m' \end{cases} \qquad m, m' = 1, 2 \tag{4.5–48}$$

where ρ is the complex correlation between the lowpass equivalent signals. The detector bases its decision on the envelopes $|\boldsymbol{r}_l \cdot \boldsymbol{s}_{1l}|$ and $|\boldsymbol{r}_l \cdot \boldsymbol{s}_{2l}|$, which are correlated (statistically dependent). Assuming that $s_1(t)$ is transmitted, these envelopes are given by

$$\begin{aligned} R_1 &= |\boldsymbol{r}_l \cdot \boldsymbol{s}_{1l}| = |2\mathcal{E}_s e^{j\phi} + \boldsymbol{n}_l \cdot \boldsymbol{s}_{1l}| \\ R_2 &= |\boldsymbol{r}_l \cdot \boldsymbol{s}_{2l}| = |2\mathcal{E}_s \rho e^{j\phi} + \boldsymbol{n}_l \cdot \boldsymbol{s}_{2l}| \end{aligned} \tag{4.5–49}$$

We note that since we are interested in the magnitudes of $2\mathcal{E}_s e^{j\phi} + \boldsymbol{n}_l \cdot \boldsymbol{s}_{1l}$ and $2\mathcal{E}_s \rho e^{j\phi} + \boldsymbol{n}_l \cdot \boldsymbol{s}_{2l}$, the effect of $e^{j\phi}$ can be absorbed in the noise component which is circular, and such a phase rotation would not affect its statistics. From above it is seen that R_1 is a Ricean random variable with parameters $s_1 = 2\mathcal{E}_s$ and $\sigma^2 = 2\mathcal{E}_s N_0$, and R_2 is a Ricean random variable with parameters $s_2 = 2\mathcal{E}_s|\rho|$ and $\sigma_2 = 2\mathcal{E}_s N_0$. These two random variables are *dependent* since the signals are not orthogonal and hence noise projections are statistically dependent.

Since R_1 and R_2 are statistically dependent, the probability of error may be obtained by evaluating the double integral

$$P_b = P(R_2 > R_1) = \int_0^\infty \int_{x_1}^\infty p(x_1, x_2)\, dx_1\, dx_2 \tag{4.5–50}$$

where $p(x_1, x_2)$ is the joint PDF of the envelopes R_1 and R_2. This approach was first used by Helstrom (1955), who determined the joint PDF of R_1 and R_2 and evaluated the double integral in Equation 4.5–50.

An alternative approach is based on the observation that the probability of error may also be expressed as

$$P_b = P\left(R_2 > R_1\right) = P\left(R_2^2 > R_1^2\right) = P\left(R_2^2 - R_1^2 > 0\right) \tag{4.5–51}$$

But $R_2^2 - R_1^2$ is a special case of a general quadratic form in complex-valued Gaussian random variables, treated later in Appendix B. For the special case under consideration, the derivation yields the error probability in the form

$$P_b = Q_1(a, b) - \frac{1}{2} e^{-\frac{a^2+b^2}{2}} I_0(ab) \tag{4.5–52}$$

where

$$\begin{aligned} a &= \sqrt{\frac{\mathcal{E}_b}{2N_0}\left(1 - \sqrt{1 - |\rho|^2}\right)} \\ b &= \sqrt{\frac{\mathcal{E}_b}{2N_0}\left(1 + \sqrt{1 - |\rho|^2}\right)} \end{aligned} \tag{4.5–53}$$

and $Q_1(a, b)$ is the Marcum Q function defined in Equations 2.3–37 and 2.3–38 and $I_0(x)$ is the modified Bessel function of order zero. Substituting Equation 4.5–53 into Equation 4.5–52 yields

$$P_b = Q_1(a, b) - \frac{1}{2} e^{-\frac{\mathcal{E}_b}{2N_0}} I_0\left(\frac{\mathcal{E}_b}{2N_0}|\rho|\right) \tag{4.5–54}$$

The error probability P_b is illustrated in Figure 4.5–3 for several values of $|\rho|$; P_b is minimized when $\rho = 0$, that is, when the signals are orthogonal. For this case, $a = 0, b = \sqrt{\mathcal{E}_b/N_0}$, and Equation 4.5–54 reduces to

$$P_b = Q_1\left(0, \sqrt{\frac{\mathcal{E}_b}{N_0}}\right) - \frac{1}{2} e^{-\mathcal{E}_b/2N_0} \tag{4.5–55}$$

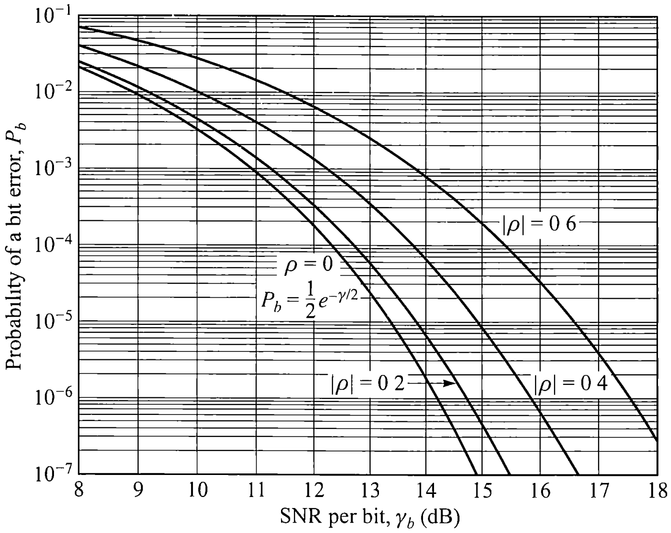


FIGURE 4.5–3
Probability of error for noncoherent detection of binary FSK.

From the properties of $Q_1(a, b)$ in Equation 2.3–39, it follows that

$$Q_1\left(0, \sqrt{\frac{\mathcal{E}_b}{N_0}}\right) = e^{-\frac{\mathcal{E}_b}{2N_0}} \qquad (4.5\text{–}56)$$

Substitution of these relations into Equation 4.5–54 yields the desired result given previously in Equation 4.5–45. On the other hand, when $|\rho| = 1$, $a = b = \sqrt{\frac{\mathcal{E}_b}{2N_0}}$ and by using Equation 2.3–38 the error probability in Equation 4.5–52 becomes $P_b = \frac{1}{2}$, as expected.

4.5–5 Differential PSK (DPSK)

We have seen in Section 4.3–2 that in order to compensate for phase ambiguity of $\frac{2\pi}{M}$, which is a result of carrier tracking by phase-locked loops (PLLs), differentially encoded PSK is used. In differentially encoded PSK, the information sequence determines the relative phase, or phase transition, between adjacent symbol intervals. Since in differential PSK the information is in the phase transitions and not in the absolute phase, the phase ambiguity from a PLL cancels between the two adjacent intervals and will have no effect on the performance of the system. The performance of the system is only slightly degraded due to the tendency of errors to occur in pairs, and the overall error probability is twice the error probability of a PSK system.

A differentially encoded phase-modulated signal also allows another type of demodulation that does not require the estimation of the carrier phase. Therefore, this type of demodulation/detection of differentially encoded PSK is classified as noncoherent detection. Since the information is in the phase transition, we have to do the detection over a period of two symbols. The vector representation of the lowpass equivalent of the mth signal over a period of two symbol intervals is given by

$$\boldsymbol{s}_{ml} = \left(\sqrt{2\mathcal{E}_s} \quad \sqrt{2\mathcal{E}_s}e^{j\theta_m}\right), \qquad 1 \le m \le M \tag{4.5–57}$$

where $\theta_m = \frac{2\pi(m-1)}{M}$ is the phase transition corresponding to the mth message. When $\boldsymbol{s}_{ml}$ is transmitted, the vector representation of the lowpass equivalent of the received signal on the corresponding two-symbol period is given by

$$\boldsymbol{r}_l = (r_1 \quad r_2) = \left(\sqrt{2\mathcal{E}_s} \quad \sqrt{2\mathcal{E}_s}e^{j\theta_m}\right)e^{j\phi} + (n_{1l} \quad n_{2l}), \qquad 1 \le m \le M \tag{4.5–58}$$

where n_{1l} and n_{2l} are two complex-valued, zero-mean, circular Gaussian random variables each with variance $2N_0$ (variance N_0 for real and imaginary components) and ϕ is the random phase due to noncoherent detection. The key assumption in this demodulation-detection scheme is that the phase offset ϕ remains the same over adjacent signaling periods. The optimal noncoherent receiver uses Equation 4.5–22 for optimal detection. We have

$$\begin{aligned}
\hat{m} &= \arg\max_{1\le m\le M} |\boldsymbol{r}_l \cdot \boldsymbol{s}_{ml}| \\
&= \arg\max_{1\le m\le M} \sqrt{2\mathcal{E}_s}\left|r_1 + r_2 e^{-j\theta_m}\right| \\
&= \arg\max_{1\le m\le M} \left|r_1 + r_2 e^{-j\theta_m}\right|^2 \\
&= \arg\max_{1\le m\le M} \left(|r_1|^2 + |r_2|^2 + 2\,\mathrm{Re}\left[r_1^* r_2 e^{-j\theta_m}\right]\right) \\
&= \arg\max_{1\le m\le M} \mathrm{Re}\left[r_1^* r_2 e^{-j\theta_m}\right] \\
&= \arg\max_{1\le m\le M} |r_1 r_2| \cos\left(\angle r_2 - \angle r_1 - \theta_m\right) \\
&= \arg\max_{1\le m\le M} \cos\left(\angle r_2 - \angle r_1 - \theta_m\right) \\
&= \arg\min_{1\le m\le M} \left|\angle r_2 - \angle r_1 - \theta_m\right|
\end{aligned} \tag{4.5–59}$$

Note that $\alpha = \angle r_2 - \angle r_1$ is the phase difference of the received signal in two adjacent intervals. The receiver computes this phase difference and compares it with $\theta_m = \frac{2\pi}{M}(m-1)$ for all $1 \le m \le M$ and selects the m for which θ_m is closest to α, thus maximizing $\cos(\alpha - \theta_m)$. A differentially encoded PSK signal that uses this method for demodulation detection is called *differential PSK* (DPSK). This method of detection has lower complexity in comparison with coherent detection of PSK signals and can be used in situations where the assumption that ϕ remains constant over two-symbol intervals is valid. As we see below, there is a performance penalty in employing this detection method.

The block diagram for the DPSK receiver is illustrated in Figure 4.5–4. In this block diagram $g(t)$ represents the baseband pulse used for phase modulation, T_s is the symbol interval, the block with the $\angle$ symbol is a phase detector, and the block with T_s introduces a delay equal to the symbol interval T_s.

Performance of Binary DPSK In binary DPSK the phase difference between adjacent symbols is either 0 or π, corresponding to a 0 or 1. The two lowpass equivalent

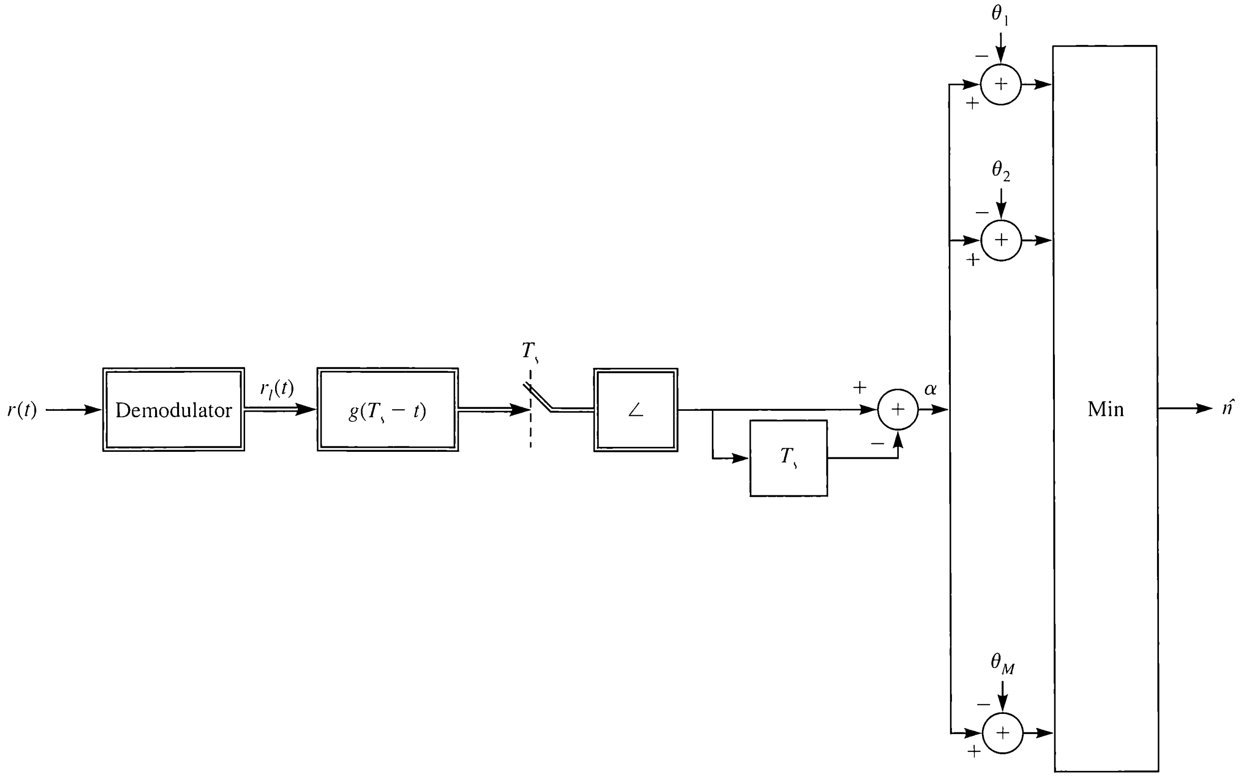


FIGURE 4.5–4
The DPSK receiver.

signals are

$$\begin{aligned} s_{1l} &= \left(\sqrt{2\mathcal{E}_s} \quad \sqrt{2\mathcal{E}_s}\right) \\ s_{2l} &= \left(\sqrt{2\mathcal{E}_s} \quad -\sqrt{2\mathcal{E}_s}\right) \end{aligned} \tag{4.5–60}$$

These two signals are noncoherently demodulated and detected using the general approach for optimal noncoherent detection. It is clear that the two signals are orthogonal on an interval of length $2T_s$. Therefore, the error probability can be obtained from the expression for the error probability of binary orthogonal signaling given in Equation 4.5–45. The difference is that the energy in each of the signals $s_1(t)$ and $s_2(t)$ is $2\mathcal{E}_s$. This is seen easily from Equation 4.5–60 which shows that the energy in lowpass equivalents is $4\mathcal{E}_s$. Therefore,

$$\begin{aligned} P_b &= \frac{1}{2} e^{-\frac{2\mathcal{E}_s}{2N_0}} \\ &= \frac{1}{2} e^{-\frac{\mathcal{E}_b}{N_0}} \end{aligned} \tag{4.5–61}$$

This is the bit error probability for binary DPSK. Comparing this result with coherent detection of BPSK where the error probability is given by

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{4.5–62}$$

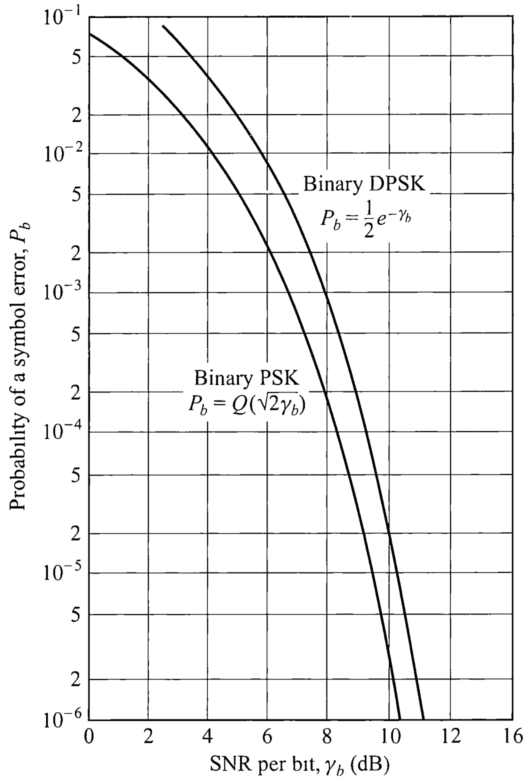


FIGURE 4.5–5
Probability of error for binary PSK and DPSK.

we observe that by the inequality $Q(x) \leq \frac{1}{2}e^{-x^2/2}$, we have

$$P_{b,\text{coh}} \leq P_{b,\text{noncoh}} \tag{4.5–63}$$

as expected. This is similar to the result we previously had for coherent and noncoherent detection of binary orthogonal FSK. Here again the difference between the performance of BPSK with coherent detection and binary DPSK at high SNRs is less than 0.8 dB. The plots given in Figure 4.5–5 compare the performance of coherently detected BPSK with binary DPSK.

Performance of DQPSK Differential QPSK is similar to binary DPSK, except that the phase difference between adjacent symbol intervals depends on two information bits ($k = 2$) and is equal to 0, $\frac{\pi}{2}$, π, and $\frac{3\pi}{2}$ for 00, 01, 11, and 10, respectively, when Gray coding is employed. Assuming that the transmitted binary sequence is 00, corresponding to a phase shift of zero in two adjacent intervals, the lowpass equivalent of the received signal over two-symbol intervals with noncoherent demodulation is given by

$$\boldsymbol{r}_l = (r_1 \quad r_2) = \left(\sqrt{2\mathcal{E}_s} \quad \sqrt{2\mathcal{E}_s}\right)e^{j\phi} + (n_1 \quad n_2) \tag{4.5–64}$$

where n_1 and n_2 are independent, zero-mean, circular, complex Gaussian random variables each with variance $2N_0$ (variance N_0 per real and complex components). The optimal decision region for 00 is given by Equation 4.5–59 as

$$D_{00} = \left\{\boldsymbol{r}_l : \operatorname{Re}\left[r_1^* r_2\right] > \operatorname{Re}\left[r_1^* r_2 e^{-j\frac{m\pi}{2}}\right], \quad \text{for } m = 1, 2, 3\right\} \tag{4.5–65}$$

where $r_1 = \sqrt{2\mathcal{E}_s}e^{j\phi} + n_1$ and $r_2 = \sqrt{2\mathcal{E}_s}e^{j\phi} + n_2$. We note that $r_1^* r_2$ does not depend on ϕ. The error probability is the probability that the received vector $\boldsymbol{r}_l$ does not belong to D_{00}. As seen from Equation 4.5–65, this probability depends on the product of two complex Gaussian random variables r_1^* and r_2. A general form of this problem, where general quadratic forms of complex Gaussian random variables are considered, is given in Appendix B. Using the result of Appendix B we can show that the bit error probability for DQPSK, when Gray coding is employed, is given by

$$P_b = Q_1(a, b) - \frac{1}{2} I_0(ab) e^{-\frac{a^2+b^2}{2}} \tag{4.5–66}$$

where $Q_1(a, b)$ is the Marcum Q function defined by Equations 2.3–37 and 2.3–38, $I_0(x)$ is the modified Bessel function of order zero, defined by Equations 2.3–32 to 2.3–34, and the parameters a and b are defined as

$$\begin{aligned} a &= \sqrt{\frac{2\mathcal{E}_b}{N_0}\left(1 - \sqrt{\frac{1}{2}}\right)} \\ b &= \sqrt{\frac{2\mathcal{E}_b}{N_0}\left(1 + \sqrt{\frac{1}{2}}\right)} \end{aligned} \tag{4.5–67}$$

Figure 4.5–6 illustrates the probability of a binary digit error for two- and four-phase DPSK and coherent PSK signaling obtained from evaluating the exact formulas derived in this section. Since binary DPSK is only slightly inferior to binary PSK at large SNR,

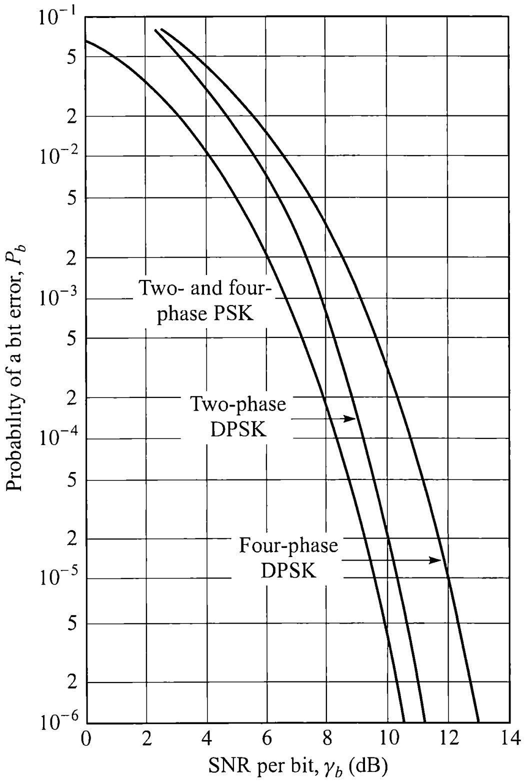


FIGURE 4.5–6
Probability of bit error for binary and four-phase PSK and DPSK.

and DPSK does not require an elaborate method for estimating the carrier phase, it is often used in digital communication systems. On the other hand, four-phase DPSK is approximately 2.3 dB poorer in performance than four-phase PSK at large SNR. Consequently the choice between these two four-phase systems is not as clear-cut. One must weigh the 2.3-dB loss against the reduction in implementation complexity.

4.6
A COMPARISON OF DIGITAL SIGNALING METHODS

The digital modulation methods described in the previous sections can be compared in a number of ways. For example, one can compare them on the basis of the SNR required to achieve a specified probability of error. However, such a comparison would not be very meaningful, unless it were made on the basis of some constraint, such as a fixed data rate of transmission or, equivalently, on the basis of a fixed bandwidth. We have already studied two major classes of signaling methods, i.e., bandwidth and power-efficient signaling in Sections 4.3 and 4.4, respectively.

The criterion for power efficiency of a signaling scheme is the SNR per bit that is required by that scheme to achieve a certain error probability. The error probability that is usually considered for comparison of various signaling schemes is $P_e = 10^{-5}$. The $\gamma_b = \frac{\mathcal{E}_b}{N_0}$ required by a signaling scheme to achieve an error probability of 10^{-5} is a criterion for power efficiency of that scheme. Systems requiring lower γ_b to achieve this error probability are more power-efficient.

To measure the bandwidth efficiency, we define a parameter r, called the *spectral bit rate,* or the *bandwidth efficiency*, as the ratio of bit rate of the signaling scheme to the bandwidth of it, i.e.,

$$r = \frac{R}{W} \quad \text{b/s/Hz} \tag{4.6–1}$$

A system with larger r is a more bandwidth-efficient system since it can transmit at a higher bit rate in each hertz of bandwidth. The parameters r and γ_b defined above are the two criteria we use for comparison of power and bandwidth efficiency of different modulation schemes. Clearly, a good system is the one that at a given γ_b provides the highest r, or at a given r requires the least γ_b.

The relation between γ_b and the error probability for individual systems was discussed in detail for different signaling schemes in the previous sections. From the expressions for error probability of various systems derived earlier in this chapter, it is easy to determine what γ_b is required to achieve an error probability of 10^{-5} in each system. In this section we discuss the relation between the bandwidth efficiency and the main parameters of a given signaling scheme.

4.6–1 Bandwidth and Dimensionality

The sampling theorem states that in order to reconstruct a signal with bandwidth W, we need to sample this signal at a rate of at least $2W$ samples per second. In other words, this signal has $2W$ degrees of freedom (dimensions) per second. Therefore, the dimensionality of signals with bandwidth W and duration T is $N = 2WT$. Although this intuitive reasoning is sufficient for our development, this statement is not precise.

It is a well-known fact, that follows from the theory of entire functions, that the only signal that is both time- and bandwidth-limited is the trivial signal $x(t) = 0$. All other signals have either infinite bandwidth and/or infinite duration. In spite of this fact, all practical signals are approximately time- and bandwidth-limited. Recall that a real signal $x(t)$ has an energy $\mathcal{E}_x$ given by

$$\mathcal{E}_x = \int_{-\infty}^{\infty} x^2(t)\,dt = \int_{-\infty}^{\infty} |X(f)|^2\,df \tag{4.6–2}$$

Here we focus on time-limited signals that are nearly bandwidth-limited. We assume that the support of $x(t)$, i.e., where $x(t)$ is nonzero, is the interval $[-T/2, T/2]$; and we also assume that $x(t)$ is η-bandwidth-limited to W, i.e., we assume that at most a fraction η of the energy in $x(t)$ is outside the frequency band $[-W, W]$. In other words,

$$\frac{1}{\mathcal{E}_x}\int_{-W}^{W} |X(f)|^2\,df \geq 1 - \eta \tag{4.6–3}$$

The *dimensionality theorem* stated below gives a precise account for the number of dimensions of the space of such signals $x(t)$.

The Dimensionality Theorem Consider the set of all signals $x(t)$ with support $[-T/2, T/2]$ that are η-bandwidth-limited to W. Then there exists a set of N orthonormal signals[†] $\{\phi_j(t), 1 \leq j \leq N\}$ with support $[-T/2, T/2]$ such that $x(t)$ can be ϵ-approximated by this set of orthonormal signals, i.e.,

$$\frac{1}{\mathcal{E}_x}\int_{-\infty}^{\infty} \left(x(t) - \sum_{j=1}^{N} \langle x(t), \phi_j(t)\rangle \phi_j(t) \right)^2 dt < \epsilon \tag{4.6–4}$$

where $\epsilon = 12\eta$ and $N = \lfloor 2WT + 1 \rfloor$.

From the dimensionality theorem we can see that the relation

$$N \approx 2WT \tag{4.6–5}$$

is a good approximation to the dimensionality of the space of functions that are roughly time-limited to T and band-limited to W.

The dimensionality theorem helps us to derive a relation between bandwidth and dimensionality of a signaling scheme. If the set of signals in a signaling scheme consists of M signals each with duration T_s, the signaling interval, and the approximate bandwidth of the set of signals is W, the dimensionality of the signal space is $N = 2WT_s$.

[†] Signals $\phi_j(t)$ can be expressed in terms of the prolate spheroidal wave functions

Using the relation $R_s = 1/T_s$, we have

$$W = \frac{R_s N}{2} \tag{4.6–6}$$

Since $R = R_s \log_2 M$, we conclude that

$$W = \frac{RN}{2\log_2 M} \tag{4.6–7}$$

and

$$r = \frac{R}{W} = \frac{2\log_2 M}{N} \tag{4.6–8}$$

This relation gives the bandwidth efficiency of a signaling scheme in terms of the constellation size and the dimensionality of the constellation.

In one-dimensional modulation schemes (ASK and PAM), $N = 1$ and $r = 2\log_2 M$. PAM and ASK can be transmitted as single-sideband (SSB) signals.

For two-dimensional signaling schemes such as QAM and MPSK, we have $N = 2$ and $r = \log_2 M$. It is clear from the above discussion that in MASK, MPSK, and MQAM signaling schemes the bandwidth efficiency increases as M increases. As we have seen before in all these systems, the power efficiency decreases as M is increased. Therefore, the size of constellation in these systems determines the tradeoff between power and bandwidth efficiency. These systems are appropriate where we have limited bandwidth and desire a bit rate–to–bandwidth ratio $r > 1$ and where there is sufficiently high SNR to support increases in M. Telephone channels and digital microwave radio channels are examples of such band-limited channels.

For M-ary orthogonal signaling, $N = M$ and hence Equation 4.6–8 results in

$$r = \frac{2\log_2 M}{M} \tag{4.6–9}$$

Obviously in this case as M increases, the bandwidth efficiency decreases, and for large M the system becomes very bandwidth-inefficient. Again as we had seen before in orthogonal signaling, increasing M improves the power efficiency of the system, and in fact this system is capable of achieving the Shannon limit as M increases. Here again the tradeoff between bandwidth and power efficiency is clear. Consequently, M-ary orthogonal signals are appropriate for power-limited channels that have sufficiently large bandwidth to accommodate a large number of signals. One example of such channels is the deep space communication channel.

We encounter the tradeoff between bandwidth and power efficiency in many communication scenarios. Coding techniques treated in Chapters 7 and 8 study various practical methods to achieve this tradeoff.

We will show in Chapter 6 that there exists a fundamental tradeoff between bandwidth and power efficiency. This tradeoff between r and $\mathcal{E}_b/N_0$ holds as P_e tends to zero and is given by (see Equation 6.5–49)

$$\frac{\mathcal{E}_b}{N_0} > \frac{2^r - 1}{r} \tag{4.6–10}$$

Equation 4.6–10 gives the condition under which reliable communication is possible. This relation should hold for any any communication system. As r tends to 0 (bandwidth becomes infinite), we can obtain the fundamental limit on the required $\mathcal{E}_b/N_0$ in a communication system. This limit is the -1.6 dB Shannon limit discussed before.

Figure 4.6–1 illustrates the graph of $r = R/W$ versus SNR per bit for PAM, QAM, PSK, and orthogonal signals, for the case in which the error probability is $P_M = 10^{-5}$. Shannon's fundamental limit given by Equation 4.6–10 is also plotted in this figure. Communication is, at least theoretically, possible at any point below this curve and is impossible at points above it.

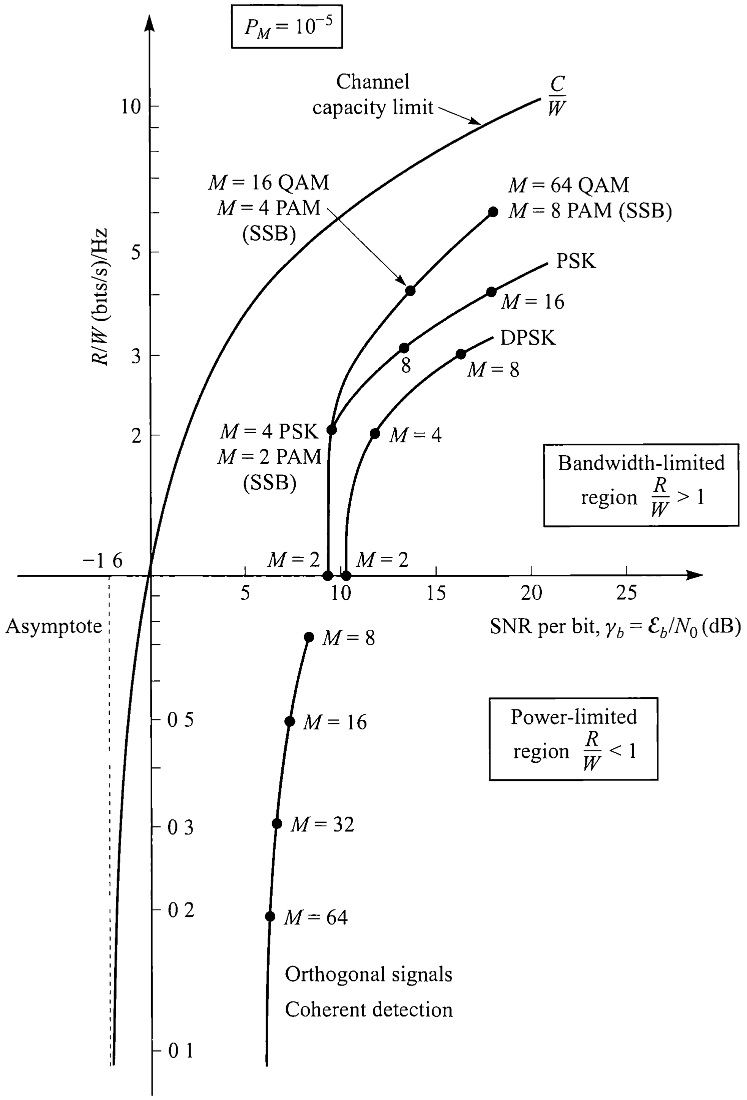


FIGURE 4.6–1

Comparison of several modulation schemes at $P_e = 10^{-5}$ symbol error probability.

4.7

LATTICES AND CONSTELLATIONS BASED ON LATTICES

In band-limited channels, when the available SNR is large, large QAM constellations are desirable to achieve high bandwidth efficiency. We have seen examples of QAM constellations in Figures 3.2–4 and 3.2–5. Figure 3.2–5 is particularly interesting since it has a useful grid-shaped repetitive pattern in two-dimensional space. Using such repetitive patterns for designing constellations is a common practice. In this approach to constellation design, a repetitive infinite grid of points and a boundary for the constellation are selected. The constellation is then defined as the set of points of the repetitive grid that are within the selected boundary. Lattices are mathematical structures that define the main properties of the repetitive grid of points used in constellation design. In this section we study properties of lattices, boundaries, and the lattice-based constellations.

4.7–1 An Introduction to Lattices

An n-dimensional *lattice* is defined as a discrete subset of $\mathbb{R}^n$ that has a group structure under ordinary vector addition. By having a group structure we mean that any two lattice points can be added and the result is another lattice point, there exists a point in the lattice denoted by $\mathbf{0}$ that when added to any lattice point $\boldsymbol{x}$ the result is $\boldsymbol{x}$ itself, and for any $\boldsymbol{x}$ there exists another point in the lattice, denoted by $-\boldsymbol{x}$, that when added to $\boldsymbol{x}$ results in $\mathbf{0}$.

With the lattice definition given above, it is clear that $\mathbb{Z}$, the set of integers, is a one-dimensional lattice. Moreover, for any $\alpha > 0$, the set $\Lambda = \alpha\mathbb{Z}$ is a one-dimensional lattice. In the plane, $\mathbb{Z}^2$, the set of all points with integer coordinates, is a two-dimensional lattice. Another example of a two-dimensional lattice, called the *hexagonal lattice*, is the set of points shown in Figure 4.7–1. These points can be written as $a(1, 0) + b\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$, where a and b are integers. The hexagonal lattice is usually denoted by $\boldsymbol{A}_2$.

In general, an n-dimensional lattice Λ can be defined in terms of n basis vectors $\boldsymbol{g}_i \in \mathbb{R}^n$, $1 \leq i \leq n$, such that any lattice point $\boldsymbol{x}$ can be written as a linear combination

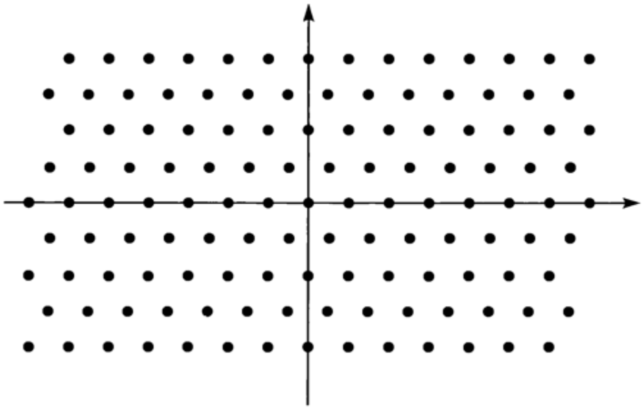


FIGURE 4.7–1
The two-dimensional hexagonal lattice.

of $\boldsymbol{g}_i$'s using integer coefficients. In other words, for any $\boldsymbol{x} \in \Lambda$,

$$\boldsymbol{x} = \sum_{i=1}^{n} a_i \boldsymbol{g}_i \tag{4.7–1}$$

where $a_i \in \mathbb{Z}$ for $1 \le i \le n$. We can also define Λ in terms of an $n \times n$ *generator matrix*, denoted by $\boldsymbol{G}$, whose rows are $\{\boldsymbol{g}_i,\ 1 \le i \le n\}$. Since the basis vectors can be selected differently, the generator matrix of a lattice is not unique. With this definition, for any $\boldsymbol{x} \in \Lambda$,

$$\boldsymbol{x} = \boldsymbol{a}\boldsymbol{G} \tag{4.7–2}$$

where $\boldsymbol{a} \in \mathbb{Z}^n$ is an n-dimensional vector with integer components. Equation 4.7–2 states that any n-dimensional lattice Λ can be viewed as a linear transformation of $\mathbb{Z}^n$ where the transformation is represented by matrix $\boldsymbol{G}$. In particular, all one-dimensional lattices can be represented as $\alpha\mathbb{Z}$ for some $\alpha > 0$.

The generator matrix of $\mathbb{Z}^2$ is $\boldsymbol{I}_2$, the 2×2 identity matrix. In general the generator matrix of $\mathbb{Z}^n$ is $\boldsymbol{I}_n$. The generator matrix of the hexagonal lattice is given by

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \tag{4.7–3}$$

Two lattices are called *equivalent* if one can be obtained from the other by a rotation, reflection, scaling, or combination of these operations. Rotation and reflection operations are represented by *orthogonal matrices*. Orthogonal matrices are matrices whose columns constitute a set of orthonormal vectors. If $\boldsymbol{A}$ is an orthogonal matrix, then $\boldsymbol{A}\boldsymbol{A}^t = \boldsymbol{A}^t\boldsymbol{A} = \boldsymbol{I}$. In general, any operation of the form $\alpha\boldsymbol{G}$ on the lattice, where $\alpha > 0$ and $\boldsymbol{G}$ is orthogonal, results in an equivalent lattice. For instance, the lattice with the generator matrix

$$\boldsymbol{G} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix} \tag{4.7–4}$$

is obtained from $\mathbb{Z}^2$ by a rotation of 45°; therefore it is equivalent to $\mathbb{Z}^2$. Note that $\boldsymbol{G}\boldsymbol{G}^t = \boldsymbol{I}$. If after rotation the resulting lattice is scaled by $\sqrt{2}$, the overall generator matrix will be

$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \tag{4.7–5}$$

This lattice is the set of points in $\mathbb{Z}^2$ for which the sum of the two coordinates is even. This lattice is also equivalent to $\mathbb{Z}^2$. Matrix $\boldsymbol{G}$ in Equation 4.7–5, which represents a rotation of 45° and a scaling of $\sqrt{2}$, is usually denoted by R. Therefore, $R\mathbb{Z}^2$ denotes the lattice of all integer coordinate points in the plane with an even sum of coordinates. It can be easily verified that $R^2\mathbb{Z}^2 = 2\mathbb{Z}^2$.

Translating (shifting) a lattice by a vector $\boldsymbol{c}$ is denoted by $\Lambda + \boldsymbol{c}$, and the result, in general, is not a lattice because under a general translation there is no guarantee that $\boldsymbol{0}$ will be a member of the translated lattice. However, if the translation vector is a lattice

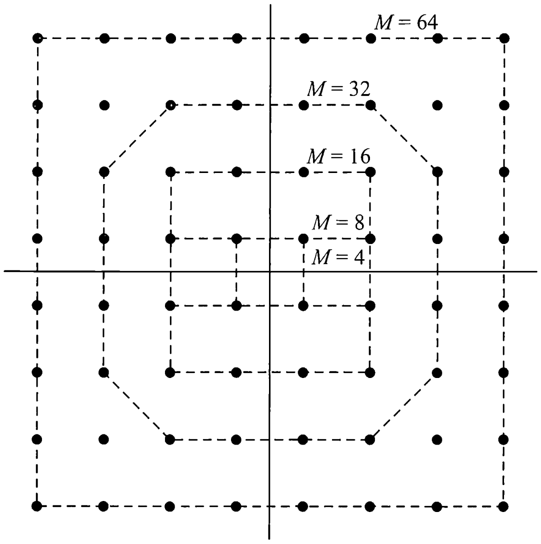


FIGURE 4.7–2
QAM constellation.

point, i.e., if $\boldsymbol{c} \in \Lambda$, then the result of translation is the original lattice. From this we conclude that any point in the lattice is similar to any other point, in the sense that all points of the lattice have the same number of lattice points at a given distance. Although translation of a lattice is not a lattice in general, the result is congruent to the original lattice with the same geometric properties. Translation of lattices is frequently used to generate energy-efficient constellations. Note that the QAM constellations shown in Figure 4.7–2 consist of points in a translated version of $\mathbb{Z}^2$ where the shift vector is $\left(\frac{1}{2}, \frac{1}{2}\right)$; i.e., the constellation points are subsets of $\mathbb{Z}^2 + \left(\frac{1}{2}, \frac{1}{2}\right)$.

In addition to rotation, reflection, scaling, and translation of lattices, we introduce the notion of the M-fold Cartesian product of lattice Λ. The M-fold Cartesian product of Λ is another lattice, denoted by Λ^M, whose elements are (Mn)-dimensional vectors $(\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_M)$ where each $\boldsymbol{\lambda}_j$ is in Λ. We observe that $\mathbb{Z}^n$ is the n-fold Cartesian product of $\mathbb{Z}$.

The *minimum distance* $d_{\min}(\Lambda)$ of a lattice Λ is the minimum Euclidean distance between any two lattice points; and the *kissing number*, or the *multiplicity*, denoted by $N_{\min}(\Lambda)$, is the number of points in the lattice that are at minimum distance from a given lattice point. If n-dimensional spheres with radius $\frac{d_{\min}(\Lambda)}{2}$ are centered at lattice points, the kissing number is the number of spheres that touch one of these spheres. For the hexagonal lattice $d_{\min}(\boldsymbol{A}_2) = 1$ and $N_{\min}(\boldsymbol{A}_2) = 6$. For $\mathbb{Z}^n$, we have $d_{\min}(\mathbb{Z}^n) = 1$ and $N_{\min}(\mathbb{Z}^n) = 2n$. In this lattice the nearest neighbors of $\mathbf{0}$ are points with $n - 1$ zero coordinates and one coordinate equal to ± 1.

The *Voronoi region* of a lattice point $\boldsymbol{x}$ is the set of all points in $\mathbb{R}^n$ that are closer to $\boldsymbol{x}$ than any other lattice point. The boundary of the Voronoi region of a lattice point $\boldsymbol{x}$ consists of the perpendicular bisector hyperplanes of the line segments connecting $\boldsymbol{x}$ to its nearest neighbors in the lattice. Therefore, a Voronoi region is a polyhedron bounded by $N_{\min}(\Lambda)$ hyperplanes. The Voronoi region of the point $\mathbf{0}$ in the hexagonal lattice is the hexagon shown in Figure 4.7–3. Since all points of the lattice have similar distances from other lattice points, the Voronoi regions of all lattice points are congruent. In addition, the Voronoi regions are disjoint and cover $\mathbb{R}^n$; hence the Voronoi regions of a lattice induce a partition of $\mathbb{R}^n$.

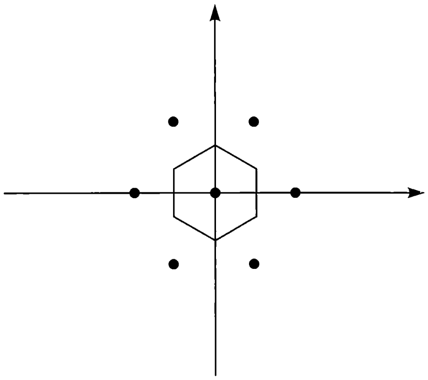


FIGURE 4.7–3
The Voronoi region in the hexagonal lattice.

The *fundamental volume* of a lattice is defined as the volume of the Voronoi region of the lattice and is denoted by $V(\Lambda)$. Since there exists one lattice point per fundamental volume, we can define the fundamental volume as the reciprocal of the number of lattice points per unit volume. It can be shown (see the book by Conway and Sloane (1999)) that for any lattice

$$V(\Lambda) = |\det(\boldsymbol{G})| \tag{4.7–6}$$

We notice that $V(\mathbb{Z}^n) = 1$ and $V(\boldsymbol{A}_2) = \frac{\sqrt{3}}{2}$.

Rotation, reflection, and translation do not change the fundamental volume, the minimum distance, or the kissing number of a lattice. Scaling a lattice Λ with generator matrix $\boldsymbol{G}$ by $\alpha > 0$ results in a lattice $\alpha\Lambda$ with generator matrix $\alpha\boldsymbol{G}$, hence

$$V(\alpha\Lambda) = |\det(\alpha\boldsymbol{G})| = \alpha^n V(\Lambda) \tag{4.7–7}$$

The minimum distance of the scaled lattice is obviously scaled by α. The kissing number of the scaled matrix is equal to the kissing number of the original lattice.

The *Hermite parameter* of a lattice is denoted by $\gamma_c(\Lambda)$ and is defined as

$$\gamma_c(\Lambda) = \frac{d^2_{\min}(\Lambda)}{[V(\Lambda)]^{\frac{2}{n}}} \tag{4.7–8}$$

This parameter has an important role in defining the *coding gain* of the lattice. It is clear that $\gamma_c(\mathbb{Z}^n) = 1$ and $\gamma_c(\boldsymbol{A}_2) = \frac{2}{\sqrt{3}} \approx 1.1547$.

Since $1/V(\Lambda)$ indicates the number of lattice points per unit volume, we conclude that among lattices with a given minimum distance, those with a higher Hermite parameter are *denser* in the sense that they have more points per unit volume. In other words, for a given $d_{\min}$, a lattice with high γ_c packs more points in unit volume. This is exactly what we need in constellation design since $d_{\min}$ determines the error probability and having more points per unit volume improves bandwidth efficiency. It is clear from above that $\boldsymbol{A}_2$ can provide 15% higher coding gain than the integer lattice $\mathbb{Z}^2$.

Some properties of $\gamma_c(\Lambda)$ are listed below. The interested reader is referred to the paper by Forney (1988) for details.

1. $\gamma_c(\Lambda)$ is a dimensionless parameter.
2. $\gamma_c(\Lambda)$ is invariant to scaling and orthogonal transformations (rotation and reflection).

3. For all M, $\gamma_c(\Lambda)$ is invariant to the M-fold Cartesian product extension of the lattice; i.e., $\gamma_c(\Lambda^M) = \gamma_c(\Lambda)$.

Multidimensional Lattices

Most lattice examples presented so far are one- or two-dimensional. We have also introduced the n-dimensional lattice $\mathbb{Z}^n$ which is an n-fold Cartesian product of $\mathbb{Z}$. In designing efficient multidimensional constellations, sometimes it is necessary to use lattices different from $\mathbb{Z}^n$. We introduce some common multidimensional lattices in this section.

We have already introduced the two-dimensional rotation and scaling matrix R as

$$R = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \tag{4.7–9}$$

This notion can be generalized to four dimensions as

$$R = \begin{bmatrix} 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \tag{4.7–10}$$

It is seen that $R^2 = 2\boldsymbol{I}_4$. Extension of this notion from 4 to $2n$ dimensions is straightforward. As a result, for any $2n$-dimensional lattice Λ we have $R^2\Lambda = 2\Lambda$. In particular $R^2\mathbb{Z}^4 = 2\mathbb{Z}^4$. Note that $R\mathbb{Z}^4$ is a lattice whose members are 4-tuples of integers in which the sum of the first two coordinates and the sum of the last two coordinates are even. Therefore $R\mathbb{Z}^4$ is a *sublattice* of $\mathbb{Z}^4$. In general, a sublattice of Λ, denoted by Λ', is a subset of points in Λ that themselves constitute a lattice. In algebraic terms, a sublattice is a subgroup of the original lattice.

We already know that $V(\mathbb{Z}^2) = 1$. From Equation 4.7–6, we have $V(R\mathbb{Z}^4) = |\det(R)| = 4$. From this it is clear that one-quarter of the points in $\mathbb{Z}^4$ belong to $R\mathbb{Z}^4$. This can also be seen from the fact that only one-quarter of points in $\mathbb{Z}^n$ have the sum of the first and the last two components both even. Therefore, we conclude that $\mathbb{Z}^4$ can be partitioned into four subsets that are all congruent to $R\mathbb{Z}^4$. We will discuss the notion of lattice partitioning and coset decomposition of lattices in Chapter 8 in the discussion of coset codes.

Another example of a multidimensional lattice is the four-dimensional *Schläfli lattice* denoted by $\boldsymbol{D}_4$. One generator matrix for this lattice is

$$\boldsymbol{G} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \tag{4.7–11}$$

This lattice represents all 4-tuples with integer coordinates in which the sum of the four coordinates is even, similar to $R\mathbb{Z}^2$ in a plane. For this lattice $V(\boldsymbol{D}_4) = |\det(\boldsymbol{G})| = 2$, and the minimum distance is the distance between points $(0, 0, 0, 0)$ and $(1, 1, 0, 0)$, thus $d_{\min}(\boldsymbol{D}_4) = \sqrt{2}$. It can be easily seen that the kissing number for this lattice is $N_{\min}(\boldsymbol{D}_4) = 24$ and

$$\gamma_c(\boldsymbol{D}_4) = \frac{d^2_{\min}(\boldsymbol{D}_4)}{[V(\boldsymbol{D}_4)]^{\frac{2}{n}}} = \frac{2}{2^{\frac{2}{4}}} = \sqrt{2} \approx 1.414 \tag{4.7–12}$$

This shows that $\mathbb{D}_4$ is approximately 41% denser than $\mathbb{Z}^4$.

Sphere Packing and Lattice Density

For any n-dimensional lattice Λ, the set of n-dimensional spheres of radius $\frac{d_{\min(\Lambda)}}{2}$ centered at all lattice points constitutes a set of nonoverlapping spheres that cover a fraction of the n-dimensional space. A measure of denseness of a lattice is the fraction of the n-dimensional space covered by these spheres. The problem of packing the space with n-dimensional spheres such that the highest fraction of the space is covered, or equivalently, packing as many possible spheres in a given volume of space, is called the *sphere packing* problem.

In the one-dimensional space, all lattices are equivalent to $\mathbb{Z}$ and the sphere packing problem becomes trivial. In this space, spheres are simply intervals of length 1 centered at lattice points. These spheres cover the entire length, and therefore the fraction of the space covered by these spheres is 1.

In Problem 4.56, it is shown that the volume of an n-dimensional sphere with radius R is given by $V_n(R) = B_n R^n$, where

$$B_n = \frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}+1\right)} \tag{4.7–13}$$

The gamma function is defined in Equation 2.3–22. In particular, note that from Equation 2.3–23 we have

$$\Gamma\left(\frac{n}{2}+1\right) = \begin{cases} \left(\frac{n}{2}\right)! & n \text{ even and positive} \\ \sqrt{\pi}\,\frac{n(n-2)(n-4)\quad 3\times 1}{2^{\frac{n+1}{2}}} & n \text{ odd and positive} \end{cases} \tag{4.7–14}$$

Substituting Equation 4.7–14 into 4.7–13 yields

$$B_n = \begin{cases} \dfrac{\pi^{\frac{n}{2}}}{\left(\frac{n}{2}\right)!} & n \text{ even} \\ \dfrac{2^n \pi^{\frac{n-1}{2}} \left(\frac{n-1}{2}\right)!}{n!} & n \text{ odd} \end{cases} \tag{4.7–15}$$

Therefore,

$$V_n(R) = \begin{cases} \dfrac{\pi^{\frac{n}{2}}}{\left(\frac{n}{2}\right)!} R^n & n \text{ even} \\ \dfrac{2^n \pi^{\frac{n-1}{2}} \left(\frac{n-1}{2}\right)!}{n!} R^n & n \text{ odd} \end{cases} \tag{4.7–16}$$

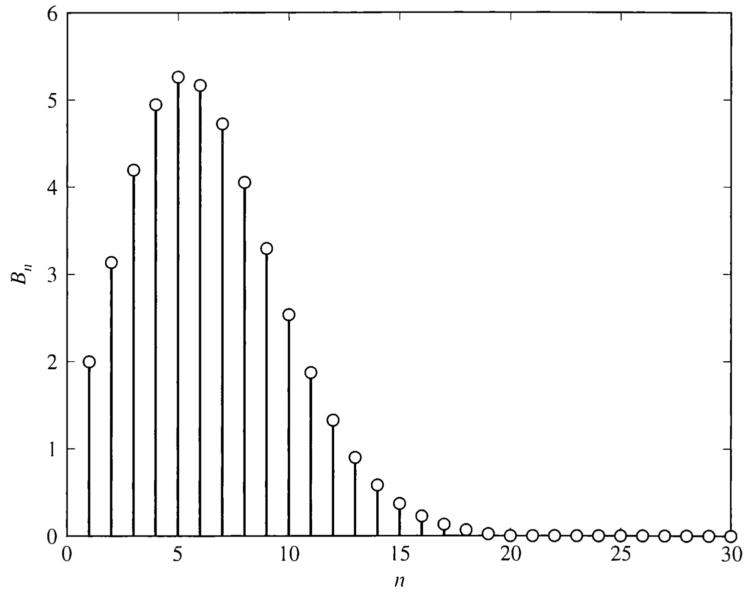


FIGURE 4.7–4
The volume of an n-dimensional sphere with radius 1.

Clearly, B_n denotes the volume of an n-dimensional sphere with radius 1. A plot of B_n for different values of n is shown in Figure 4.7–4. It is interesting to note that for large n the value of B_n goes to zero, and it has a maximum at $n = 5$.

The volume of the space that corresponds to each lattice point is $V(\Lambda)$, the fundamental volume of the lattice. We define the *density* of a lattice Λ, denoted by $\Delta(\Lambda)$, as the ratio of the volume of a sphere with radius $\frac{d_{\min}(\Lambda)}{2}$ to the fundamental volume of the lattice. This ratio is the fraction of the space covered by the spheres of radius $\frac{d_{\min}(\Lambda)}{2}$ and centered at lattice points. From this definition we have

$$\begin{aligned}
\Delta(\Lambda) &= \frac{V_n\left(\frac{d_{\min}(\Lambda)}{2}\right)}{V(\Lambda)} \\
&= \frac{B_n}{V(\Lambda)}\left(\frac{d_{\min}(\Lambda)}{2}\right)^n \\
&= \frac{B_n}{2^n}\left(\frac{d^2_{\min}(\Lambda)}{V^{\frac{2}{n}}(\Lambda)}\right)^{\frac{n}{2}} \\
&= \frac{B_n}{2^n}\,\gamma_c^{\frac{n}{2}}(\Lambda)
\end{aligned} \tag{4.7–17}$$

where we have used the definition of $\gamma_c(\Lambda)$ given in Equation 4.7–8.

EXAMPLE 4.7–1. To obtain the density of $\mathbb{Z}^2$, we note that for this lattice $n = 2$, $d_{\min} = 1$, and $V(\mathbb{Z}^2) = 1$. Substituting in Equation 4.7–17, we obtain

$$\Delta(\mathbb{Z}^n) = \frac{B_n}{V(\Lambda)}\left(\frac{d_{\min}(\Lambda)}{2}\right)^n = \pi\left(\frac{1}{2}\right)^2 = \frac{\pi}{4} = 0.7854 \tag{4.7–18}$$

For $\boldsymbol{A}_2$ we have $n = 2$, $d_{\min} = 1$, and $V(\boldsymbol{A}_2) = \frac{\sqrt{3}}{2}$. Therefore,

$$\Delta(\boldsymbol{A}_2) = \frac{B_n}{V(\Lambda)} \left(\frac{d_{\min}(\Lambda)}{2} \right)^n = \frac{\pi}{\frac{\sqrt{3}}{2}} \left(\frac{1}{2} \right)^2 = \frac{\pi}{2\sqrt{3}} = 0.9069 \tag{4.7–19}$$

This shows that $\boldsymbol{A}_2$ is denser than $\mathbb{Z}^2$.

It can be shown that among all two-dimensional lattices, $\boldsymbol{A}_2$ has the highest density. Therefore the hexagonal lattice provides the best sphere packing in the plane.

EXAMPLE 4.7–2. For $\boldsymbol{D}_4$, the Schläfli lattice, we have $n = 4$, $d_{\min}(\boldsymbol{D}_4) = \sqrt{2}$, and and $V(\boldsymbol{D}_4) = 2$. Therefore,

$$\Delta(\boldsymbol{A}_2) = \frac{B_n}{V(\Lambda)} \left(\frac{d_{\min}(\Lambda)}{2} \right)^n = \frac{\pi^2}{16} = 0.6169 \tag{4.7–20}$$

4.7–2 Signal Constellations from Lattices

A signal constellation $\mathcal{C}$ can be carved from a lattice by choosing the points of a lattice, or a shifted version of it, that are within some region $\mathcal{R}$. The signal points are therefore the intersection of the lattice points, or its shift, and region $\mathcal{R}$, i.e., $\mathcal{C}(\Lambda, \mathcal{R}) = (\Lambda + \boldsymbol{a}) \cap \mathcal{R}$, where $\boldsymbol{a}$ denotes a possible shift in lattice points. For instance, in Figure 4.7–2, the points of the constellation belong to $\mathbb{Z}^2 + \left(\frac{1}{2}, \frac{1}{2}\right)$, and the region $\mathcal{R}$ is either a square or a cross-shaped region depending on the constellation size. For $M = 4, 16, 64$, $\mathcal{R}$ is a square; and for $M = 8, 32$ it has a cross shape. The constellation size M is the number of lattice (or shifted lattice) points within the boundary. Since $V(\Lambda)$ is the reciprocal of the number of lattice points per unit volume, we conclude that if the volume of the region $\mathcal{R}$, denoted by $V(\mathcal{R})$, is much larger than $V(\Lambda)$, then

$$M \approx \frac{V(\mathcal{R})}{V(\Lambda)} \tag{4.7–21}$$

The average energy of a constellation with equiprobable messages is

$$\mathcal{E}_{\text{avg}} = \frac{1}{M} \sum_{m=1}^{M} \|\boldsymbol{x}_m\|^2 \tag{4.7–22}$$

For a large constellation we can use the *continuous approximation* by assuming that the probability is uniformly distributed on the region $\mathcal{R}$, and by finding the *second moment of the region* as

$$\mathcal{E}(\mathcal{R}) = \frac{1}{V(\mathcal{R})} \int_{\mathcal{R}} \|\boldsymbol{x}\|^2 \, d\boldsymbol{x} \tag{4.7–23}$$

For large values of M, $\mathcal{E}(\mathcal{R})$ is quite close to $\mathcal{E}_{\text{avg}}$. Table 4.7–1 gives values of $\mathcal{E}(\mathcal{R})$ and $\mathcal{E}_{\text{avg}}$ for $M = 16, 64, 256$ for a square constellation. The last column of this table gives the relative error in substituting the average energy with the continuous approximation.

TABLE 4.7–1
Average Energy and Its Continuous Approximation for Square Constellations

M	$\mathcal{E}_{\text{avg}}$	$\mathcal{E}(\mathcal{R})$	$\frac{\mathcal{E}(\mathcal{R})-\mathcal{E}_{\text{avg}}}{\mathcal{E}(\mathcal{R})}$
16	$\frac{5}{2}$	$\frac{8}{3}$	0.06
64	$\frac{21}{2}$	$\frac{32}{3}$	0.015
256	$\frac{85}{2}$	$\frac{128}{3}$	0.004

To be able to compare an n-dimensional constellation $\mathcal{C}$ with QAM, we define the *average energy per two dimensions* as

$$\mathcal{E}_{\text{avg/2D}}(\mathcal{C}) = \frac{2}{n}\mathcal{E}_{\text{avg}} = \frac{2}{nM}\sum_{m\in\mathcal{C}} \|\boldsymbol{x}_m\|^2 \tag{4.7–24}$$

Using the continuous approximation, the average energy per two dimensions can be well approximated by

$$\mathcal{E}_{\text{avg/2D}} \approx \frac{2}{nV(\mathcal{R})}\int_{\mathcal{R}} \|\boldsymbol{x}\|^2\, d\boldsymbol{x} \tag{4.7–25}$$

Error Probability and Constellation Figure of Merit

In a lattice-based constellation, each signal point has $N_{\min}$ nearest neighbors; therefore at high SNRs we have

$$P_e \approx N_{\min} Q\left(\sqrt{\frac{d^2_{\min}}{2N_0}}\right) \tag{4.7–26}$$

An efficient constellation provides large $d_{\min}$ at a given average energy. To study and compare the efficiency of different constellations, we express the error probability as

$$P_e \approx N_{\min} Q\left(\sqrt{\frac{d^2_{\min}}{2\mathcal{E}_{\text{avg/2D}}}\cdot\frac{\mathcal{E}_{\text{avg/2D}}}{N_0}}\right) \tag{4.7–27}$$

The term $\frac{\mathcal{E}_{\text{avg/2D}}}{N_0}$ represents the average SNR per two dimensions and is denoted by $\text{SNR}_{\text{avg/2D}}$. The numerator of $\text{SNR}_{\text{avg/2D}}$ is the average signal energy per two dimensions, and its denominator is the noise power per two dimensions. If we define the *constellation figure of merit* (CFM) as

$$\text{CFM}(\mathcal{C}) = \frac{d^2_{\min}(\mathcal{C})}{\mathcal{E}_{\text{avg/2D}}(\mathcal{C})} \tag{4.7–28}$$

where $\mathcal{E}_{\text{avg/2D}}(\mathcal{C})$ is given by Equation 4.7–24, we can express the error probability from Equation 4.7–27 as

$$P_e \approx N_{\min} Q\left(\sqrt{\frac{\text{CFM}(\mathcal{C})}{2} \cdot \frac{\mathcal{E}_{\text{avg/2D}}}{N_0}}\right) = N_{\min} Q\left(\sqrt{\frac{\text{CFM}(\mathcal{C})}{2} \cdot \text{SNR}_{\text{avg/2D}}}\right) \tag{4.7–29}$$

Clearly the constellation figure of merit determines the coefficient by which the $\mathcal{E}_{\text{avg/2D}}(\mathcal{C})$ is scaled in the expression of error probability.

For a square QAM constellation from Equation 3.2–41 we have

$$d^2_{\min} = \frac{6\mathcal{E}_{\text{avg}}}{M-1} \tag{4.7–30}$$

Therefore,

$$\text{CFM} = \frac{6}{M-1} \tag{4.7–31}$$

Note that from Equation 4.3–30 we have

$$P_e \approx 4Q\left(\sqrt{\frac{3}{M-1}\frac{\mathcal{E}_{\text{avg}}}{N_0}}\right) = 4Q\left(\sqrt{\frac{\text{CFM}}{2}\frac{\mathcal{E}_{\text{avg}}}{N_0}}\right) \tag{4.7–32}$$

which is in agreement with Equation 4.7–29. Also note that in a square QAM constellation, for large M we can write

$$\text{CFM} \approx \frac{6}{M} = \frac{6}{2^k} \tag{4.7–33}$$

where k denotes the number of bits per two dimensions.

Coding and Shaping Gains

In Problem 4.57 we consider a constellation $\mathcal{C}$ based on the intersection of the shifted lattice $\mathbb{Z}^n + \left(\frac{1}{2}, \frac{1}{2}, \dots, \frac{1}{2}\right)$ and the boundary region $\mathcal{R}$ defined as an n-dimensional hypercube centered at the origin with side length L. In this problem it is shown that when n is even, and $L = 2^\ell$ is a power of 2, the number of bits per two dimensions, denoted by β, is equal to $2\ell + 2$, and CFM($\mathcal{C}$) is approximated by

$$\text{CFM}(\mathcal{C}) \approx \frac{6}{2^\beta} \tag{4.7–34}$$

which is equal to what we obtained for a square QAM. Since the $\mathbb{Z}^n$ with the cubic boundary is the simplest possible n-dimensional constellation, its CFM is taken as the baseline CFM to which the CFMs of other constellations are compared. This *baseline constellation figure of merit* is denoted by CFM$_0$. Note that in an n-dimensional constellation of size M, the number of bits per two dimensions is

$$\beta = \frac{2}{n}\log_2 M \tag{4.7–35}$$

Hence,

$$2^{\beta} = M^{\frac{2}{n}} \tag{4.7–36}$$

From this and Equation 4.7–21, we have

$$2^{\beta} \approx \left[\frac{V(\mathcal{R})}{V(\Lambda)}\right]^{\frac{2}{n}} \tag{4.7–37}$$

Using this result in Equation 4.7–34 gives the value of the baseline constellation figure of merit as

$$\text{CFM}_0 = \frac{6}{2^{\beta}} \approx 6\left[\frac{V(\Lambda)}{V(\mathcal{R})}\right]^{\frac{2}{n}} \tag{4.7–38}$$

From Equations 4.7–28 and 4.7–38 we have

$$\frac{\text{CFM}(\mathcal{C})}{\text{CFM}_0} \approx \frac{d^2_{\text{min}}}{[V(\Lambda)]^{\frac{2}{n}}} \times \frac{[V(\mathcal{R})]^{\frac{2}{n}}}{6\mathcal{E}_{\text{avg/2D}}} \tag{4.7–39}$$

Now we define the *shaping gain of region* $\mathcal{R}$ as

$$\begin{aligned}\gamma_s(\mathcal{R}) &= \frac{[V(\mathcal{R})]^{\frac{2}{n}}}{6\mathcal{E}_{\text{avg/2D}}} \\ &\approx \frac{n[V(\mathcal{R})]^{1+\frac{2}{n}}}{12\int_{\mathcal{R}} \|\boldsymbol{x}\|^2\, d\boldsymbol{x}}\end{aligned} \tag{4.7–40}$$

where in the last step we used Equation 4.7–25. It can be shown that the shaping gain is independent of scaling and orthogonal transformations of the region $\mathcal{R}$. It can also be shown that $\gamma_s\left(\mathcal{R}^M\right) = \gamma_s(\mathcal{R})$, where $\mathcal{R}^M$ denotes the M-fold Cartesian product of the boundary region $\mathcal{R}$. From these, and the properties of $\gamma_c(\Lambda)$, it is clear that scaling, orthogonal transformation, and Cartesian product of Λ and $\mathcal{R}$ have no effect on the figure of merit of the constellation based on Λ and $\mathcal{R}$.

From Equation 4.7–39 we have

$$\text{CFM}(\mathcal{C}) \approx \text{CFM}_0 \cdot \gamma_c(\Lambda) \cdot \gamma_s(\mathcal{R}) \tag{4.7–41}$$

This relation shows that the relative gain of a given constellation over the baseline constellation can be viewed as the product of two independent terms, namely, the *fundamental coding gain of the lattice*, denoted by $\gamma_c(\Lambda)$ and given by Equation 4.7–8, and the *shaping gain of region* $\mathcal{R}$, denoted by $\gamma_s(\mathcal{R})$ and given in Equation 4.7–40. The fundamental coding gain depends on the choice of the lattice. Choosing a dense lattice with high coding gain that provides large minimum distance per unit volume, or, equivalently, requires low volume for a given minimum distance, is highly desirable and improves the performance. Similarly, the shaping gain depends only on the choice of the boundary of the constellation, and choosing a region $\mathcal{R}$ with high shaping gain improves the power efficiency of the constellation and results in improved performance of the system.

In Problem 4.57 it is shown that if $\mathcal{R}$ is an n-dimensional hypercube centered at the origin, then $\gamma_s(\mathcal{R}) = 1$.

EXAMPLE 4.7–3. For a circle of radius r, we have $V(\mathcal{R}) = \pi r^2$ and

$$\begin{aligned}\iint_{x^2+y^2\le r^2} (x^2+y^2)\,dx\,dy &= \int_0^{2\pi}\int_0^{r} z^2\, z\,dz\,d\theta \\ &= \frac{\pi}{2} r^4\end{aligned} \tag{4.7–42}$$

Therefore,

$$\begin{aligned}\gamma_s(\mathcal{R}) &= \frac{n\,[V(\mathcal{R})]^{1+\frac{2}{n}}}{12\displaystyle\int_{\mathcal{R}} \|\boldsymbol{x}\|^2\,d\boldsymbol{x}} \\ &= \frac{2(\pi r^2)^2}{6\pi r^4} \\ &= \frac{\pi}{3} \approx 1.0472 \sim 0.2 \text{ dB}\end{aligned} \tag{4.7–43}$$

Recall that $\gamma_c(\boldsymbol{A}_2) \approx 1.1547 \sim 0.62$ dB; therefore a hexagonal constellation with a circular boundary is capable of providing an asymptotic overall gain of 0.82 dB over the baseline constellation.

EXAMPLE 4.7–4. As a generalization of Example 4.7–3, let us consider the case where $\mathcal{R}$ is an n-dimensional sphere of radius R and centered at the origin. In this case

$$\begin{aligned}\int_{\mathcal{R}} \|\boldsymbol{x}\|^2\,d\boldsymbol{x} &= \int_0^R r^2\,dV_n(r) \\ &= \int_0^R r^2\,d(B_n r^n) \\ &= B_n\int_0^R n r^{n+1}\,dr \\ &= \frac{nB_n}{n+2}R^{n+2} \\ &= \frac{n}{n+2}R^2 V_n(R)\end{aligned} \tag{4.7–44}$$

Substituting this result into Equation 4.7–40 yields

$$\gamma_s(\mathcal{R}) = \frac{n+2}{12}\left(\frac{V_n^{\frac{1}{n}}(R)}{R}\right)^2 \tag{4.7–45}$$

Note that $V_n^{\frac{1}{n}}(R)$ is the length of the side of an n-dimensional cube that has a volume equal to an n-dimensional sphere of radius R. Substituting for $V_n(R)$ from Equation 4.7–16 results in

$$\gamma_s(\mathcal{R}) = \frac{(n+2)\pi}{12\left[\Gamma\left(\frac{n}{2}+1\right)\right]^{\frac{2}{n}}} \tag{4.7–46}$$

A plot of $\gamma_s(\mathcal{R})$ for an n-dimensional sphere as a function of n is shown in Figure 4.7–5.

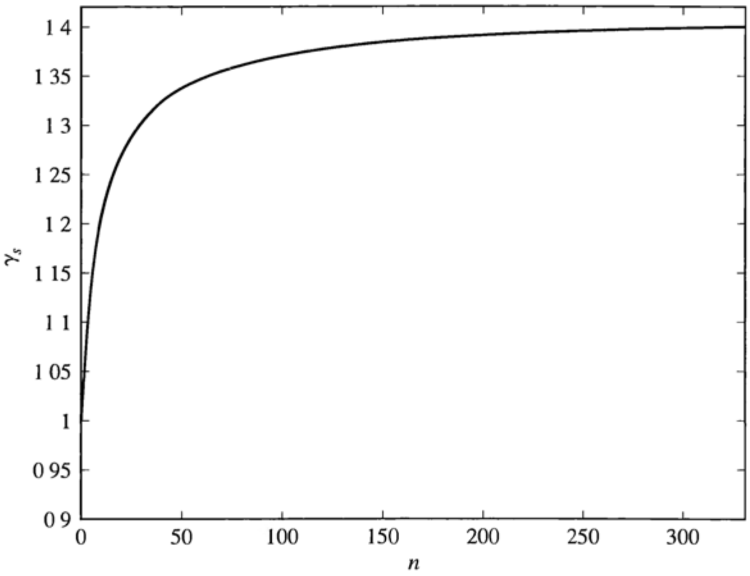


FIGURE 4.7–5
The shaping gain for an n-dimensional sphere.

It can be shown that among all possible boundaries in an n-dimensional space, spherical boundaries are the most efficient. As the dimensionality of the space increases, spherical boundaries can provide an asymptotic shaping gain of $\frac{\pi e}{6}$ which is approximately 1.423 equivalent to 1.533 dB. Therefore, 1.533 dB is the maximum gain that shaping can provide. Getting close to this bound requires high dimensional constellations. For instance, increasing the dimensionality of the space to 100 will provide a shaping gain of roughly 1.37 dB, and increasing it to 1000 provides a shaping gain of 1.5066 dB.

Unlike shaping gain, the coding gain can be increased indefinitely by using high dimensional dense lattices. However, such lattices have very large kissing numbers. The effect of large kissing numbers dramatically offsets the effect of the increased coding gain, and the overall performance of the system will remain within the bounds predicted by Shannon and discussed in Chapter 6.

4.8

DETECTION OF SIGNALING SCHEMES WITH MEMORY

When the signal has no memory, the symbol-by-symbol detector described in the preceding sections of this chapter is optimum in the sense of minimizing the probability of a symbol error. On the other hand, when the transmitted signal has memory, i.e., the signals transmitted in successive symbol intervals are interdependent, then the optimum detector is a detector that bases its decisions on observation of a sequence of received signals over successive signal intervals. In this section, we describe a maximum-likelihood sequence detection algorithm that searches for the minimum Euclidean distance path through the trellis that characterizes the memory in the transmitted signal. Another possible approach is a maximum a posteriori probability algorithm that makes decisions on a symbol-by-symbol basis, but each symbol decision is based on an observation of a sequence of received signal vectors. This approach is similar to the maximum a posteriori detection rule used for decoding turbo codes, known as the BCJR algorithm, that will be discussed in Chapter 8.

4.8–1 The Maximum Likelihood Sequence Detector

Modulation systems with memory can be modeled as finite-state machines which can be represented by a trellis, and the transmitted signal sequence corresponds to a path through the trellis. Let us assume that the transmitted signal has a duration of K symbol intervals. If we consider transmission over K symbol intervals, and each path of length K through the trellis as a message signal, then the problem reduces to the optimal detection problem discussed earlier in this chapter. The number of messages in this case is equal to the number of paths through the trellis, and a *maximum likelihood sequence detection* (MLSD) algorithm selects the most likely path (sequence) corresponding to the received signal $r(t)$ over the K signaling interval. As we have seen before, ML detection corresponds to selecting a path of K signals through the trellis such that the Euclidean distance between that path and $r(t)$ is minimized. Note that since

$$\int_0^{KT_s} |r(t) - s(t)|^2 \, dt = \sum_{k=1}^{K} \int_{(k-1)T_s}^{kT_s} |r(t) - s(t)|^2 \, dt \tag{4.8–1}$$

the optimal detection rule becomes

$$\begin{aligned} \left(\hat{\boldsymbol{s}}^{(1)}, \hat{\boldsymbol{s}}^{(2)}, \ldots, \hat{\boldsymbol{s}}^{(K)}\right) &= \underset{\left(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}, \ldots, \boldsymbol{s}^{(K)}\right) \in \Upsilon}{\arg\min} \sum_{k=1}^{K} \|\boldsymbol{r}^{(k)} - \boldsymbol{s}^{(k)}\|^2 \\ &= \underset{\left(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}, \ldots, \boldsymbol{s}^{(K)}\right) \in \Upsilon}{\arg\min} \sum_{k=1}^{K} D\left(\boldsymbol{r}^{(k)}, \boldsymbol{s}^{(k)}\right) \end{aligned} \tag{4.8–2}$$

where Υ denotes the trellis. The above argument applies to all modulation systems with memory.

As an example of the maximum-likelihood sequence detection algorithm, let us consider the NRZI signal described in Section 3.3. Its memory is characterized by the trellis shown in Figure 3.3–3. The signal transmitted in each signal interval is binary PAM. Hence, there are two possible transmitted signals corresponding to the signal points $s_1 = -s_2 = \sqrt{\mathcal{E}_b}$, where $\mathcal{E}_b$ is the energy per bit.

In searching through the trellis for the most likely sequence, it may appear that we must compute the Euclidean distance for every possible sequence. For the NRZI example, which employs binary modulation, the total number of sequences is 2^K. However, this is not the case. We may reduce the number of sequences in the trellis search by using the *Viterbi algorithm* to eliminate sequences as new data are received from the demodulator.

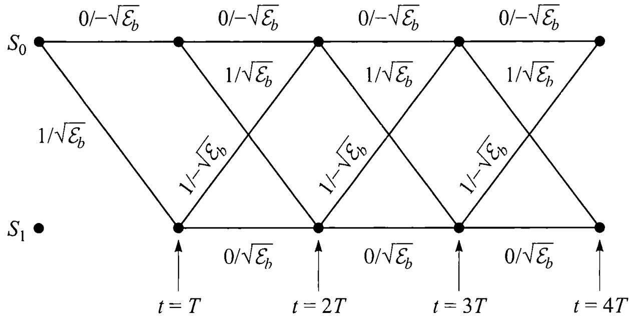


FIGURE 4.8–1
Trellis for NRZI signal.

The Viterbi algorithm is a sequential trellis search algorithm for performing ML sequence detection. It is described in Chapter 8 as a decoding algorithm for convolutional codes. We describe it below in the context of the NRZI signal detection. We assume that the search process begins initially at state S_0. The corresponding trellis is shown in Figure 4.8–1.

At time $t = T$, we receive $r_1 = s_1^{(m)} + n$ from the demodulator, and at $t = 2T$, we receive $r_2 = s_2^{(m)} + n_2$. Since the signal memory is 1 bit, which we denote by $L = 1$, we observe that the trellis reaches its regular (steady-state) form after two transitions. Thus, upon receipt of r_2 at $t = 2T$ (and thereafter), we observe that there are two signal paths entering each of the nodes and two signal paths leaving each node. The two paths entering node S_0 at $t = 2T$ correspond to the information bits (0, 0) and (1, 1) or, equivalently, to the signal points $(-\sqrt{\mathcal{E}_b}, -\sqrt{\mathcal{E}_b})$ and $(\sqrt{\mathcal{E}_b}, -\sqrt{\mathcal{E}_b})$, respectively. The two paths entering node S_1 at $t = 2T$ correspond to the information bits (0, 1) and (1, 0) or, equivalently, to the signal points $(-\sqrt{\mathcal{E}_b}, \sqrt{\mathcal{E}_b})$ and $(\sqrt{\mathcal{E}_b}, \sqrt{\mathcal{E}_b})$, respectively.

For the two paths entering node S_0, we compute the two Euclidean distance metrics

$$\begin{aligned} D_0(0,0) &= (r_1 + \sqrt{\mathcal{E}_b})^2 + (r_2 + \sqrt{\mathcal{E}_b})^2 \\ D_0(1,1) &= (r_1 - \sqrt{\mathcal{E}_b})^2 + (r_2 + \sqrt{\mathcal{E}_b})^2 \end{aligned} \tag{4.8–3}$$

by using the outputs r_1 and r_2 from the demodulator. The Viterbi algorithm compares these two metrics and discards the path having the larger (greater-distance) metric.[†] The other path with the lower metric is saved and is called the *survivor* at $t = 2T$. The elimination of one of the two paths may be done without compromising the optimality of the trellis search, because any extension of the path with the larger distance beyond $t = 2T$ will always have a larger metric than the survivor that is extended along the same path beyond $t = 2T$.

Similarly, for the two paths entering node S_1 at $t = 2T$, we compute the two Euclidean distance metrics

$$\begin{aligned} D_1(0,1) &= (r_1 + \sqrt{\mathcal{E}_b})^2 + (r_2 - \sqrt{\mathcal{E}_b})^2 \\ D_1(1,0) &= (r_1 - \sqrt{\mathcal{E}_b})^2 + (r_2 - \sqrt{\mathcal{E}_b})^2 \end{aligned} \tag{4.8–4}$$

[†] Note that, for NRZI, the reception of r_2 from the demodulator neither increases nor decreases the relative difference between the two metrics $D_0(0, 0)$ and $D_0(1, 1)$. At this point, one may ponder the implications of this observation In any case, we continue with the description of the ML sequence detection based on the Viterbi algorithm

by using the outputs r_1 and r_2 from the demodulator. The two metrics are compared, and the signal path with the larger metric is eliminated. Thus, at $t = 2T$, we are left with two survivor paths, one at node S_0 and the other at node S_1, and their corresponding metrics. The signal paths at nodes S_0 and S_1 are then extended along the two survivor paths.

Upon receipt of r_3 at $t = 3T$, we compute the metrics of the two paths entering state S_0. Suppose the survivors at $t = 2T$ are the paths (0, 0) at S_0 and (0, 1) at S_1. Then the two metrics for the paths entering S_0 at $t = 3T$ are

$$\begin{aligned} D_0(0,0,0) &= D_0(0,0) + (r_3 + \sqrt{\mathcal{E}_b})^2 \\ D_0(0,1,1) &= D_1(0,1) + (r_3 + \sqrt{\mathcal{E}_b})^2 \end{aligned} \tag{4.8–5}$$

These two metrics are compared, and the path with the larger (greater-distance) metric is eliminated. Similarly, the metrics for the two paths entering S_1 at $t = 3T$ are

$$\begin{aligned} D_1(0,0,1) &= D_0(0,0) + (r_3 - \sqrt{\mathcal{E}_b})^2 \\ D_1(0,1,0) &= D_1(0,1) + (r_3 - \sqrt{\mathcal{E}_b})^2 \end{aligned} \tag{4.8–6}$$

These two metrics are compared, and the path with the larger (greater-distance) metric is eliminated.

This process is continued as each new signal sample is received from the demodulator. Thus, the Viterbi algorithm computes two metrics for the two signal paths entering a node at each stage of the trellis search and eliminates one of the two paths at each node. The two survivor paths are then extended forward to the next state. Therefore, the number of paths searched in the trellis is reduced by a factor of 2 at each stage.

It is relatively easy to generalize the trellis search performed by the Viterbi algorithm for M-ary modulation. For example, consider a system that employs $M = 4$ signals and is characterized by the four-state trellis shown in Figure 4.8–2. We observe that each state has two signal paths entering and two signal paths leaving each node. The memory of the signal is $L = 1$. Hence, the Viterbi algorithm will have four survivors at each stage and their corresponding metrics. Two metrics corresponding to the two entering paths are computed at each node, and one of the two signal paths entering the

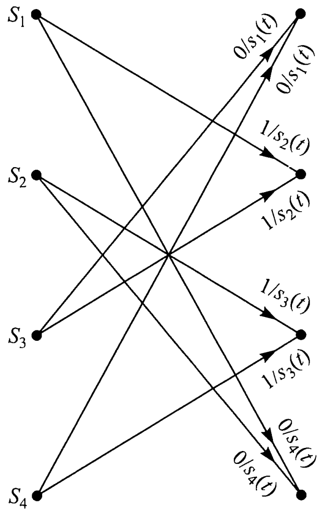


FIGURE 4.8–2
One stage of trellis diagram for delay modulation.

node is eliminated at each state of the trellis. Thus, the Viterbi algorithm minimizes the number of trellis paths searched in performing ML sequence detection.

From the description of the Viterbi algorithm given above, it is unclear how decisions are made on the individual detected information symbols given the surviving sequences. If we have advanced to some stage, say K, where $K \gg L$ in the trellis, and we compare the surviving sequences, we shall find that with high probability all surviving sequences will be identical in bit (or symbol) positions $K - 5L$ and less. In a practical implementation of the Viterbi algorithm, decisions on each information bit (or symbol) are forced after a delay of $5L$ bits (or symbols), and hence the surviving sequences are truncated to the $5L$ most recent bits (or symbols). Thus, a variable delay in bit or symbol detection is avoided. The loss in performance resulting from the sub-optimum detection procedure is negligible if the delay is at least $5L$. This approach to implementation of Viterbi algorithm is called *path memory truncation*.

EXAMPLE 4.8–1. Consider the decision rule for detecting the data sequence in an NRZI signal with a Viterbi algorithm having a delay of $5L$ bits. The trellis for the NRZI signal is shown in Figure 4.8–1. In this case, $L = 1$; hence the delay in bit detection is set to 5 bits. Hence, at $t = 6T$, we shall have two surviving sequences, one for each of the two states and the corresponding metrics $\mu_6(b_1, b_2, b_3, b_4, b_5, b_6)$ and $\mu_6(b'_1, b'_2, b'_3, b'_4, b'_5, b'_6)$. At this stage, with probability nearly equal to 1, bit b_1 will be the same as b'_1; that is, both surviving sequences will have a common first branch. If $b_1 \neq b'_1$, we may select the bit (b_1 or b'_1) corresponding to the smaller of the two metrics. Then the first bit is dropped from the two surviving sequences. At $t = 7T$, the two metrics $\mu_7(b_2, b_3, b_4, b_5, b_6, b_7)$ and $\mu_7(b'_2, b'_3, b'_4, b'_5, b'_6, b'_7)$ will be used to determine the decision on bit b_2. This process continues at each stage of the search through the trellis for the minimum-distance sequence. Thus the detection delay is fixed at 5 bits.[†]

4.9
OPTIMUM RECEIVER FOR CPM SIGNALS

We recall from Section 3.3–2 that CPM is a modulation method with memory. The memory results from the continuity of the transmitted carrier phase from one signal interval to the next. The transmitted CPM signal may be expressed as

$$s(t) = \sqrt{\frac{2\mathcal{E}}{T}} \cos[2\pi f_c t + \phi(t; \boldsymbol{I})] \tag{4.9–1}$$

where $\phi(t; \boldsymbol{I})$ is the carrier phase. The filtered received signal for an additive Gaussian noise channel is

$$r(t) = s(t) + n(t) \tag{4.9–2}$$

[†] One may have observed by now that the ML sequence detector and the symbol-by-symbol detector that ignores the memory in the NRZI signal reach the same decision Hence, there is no need for a decision delay. Nevertheless, the procedure described above applies in general.

where

$$n(t) = n_i(t)\cos 2\pi f_c t - n_q(t)\sin 2\pi f_c t \tag{4.9–3}$$

4.9–1 Optimum Demodulation and Detection of CPM

The optimum receiver for this signal consists of a correlator followed by a maximum-likelihood sequence detector that searches the paths through the state trellis for the minimum Euclidean distance path. The Viterbi algorithm is an efficient method for performing this search. Let us establish the general state trellis structure for CPM and then describe the metric computations.

Recall that the carrier phase for a CPM signal with a fixed modulation index h may be expressed as

$$\begin{aligned} \phi(t; \boldsymbol{I}) &= 2\pi h \sum_{k=-\infty}^{n} I_k q(t - kT) \\ &= \pi h \sum_{k=-\infty}^{n-L} I_k + 2\pi h \sum_{k=n-L+1}^{n} I_k q(t - kT) \\ &= \theta_n + \theta(t; \boldsymbol{I}), \qquad nT \le t \le (n+1)T \end{aligned} \tag{4.9–4}$$

where we have assumed that $q(t) = 0$ for $t < 0$, $q(t) = \frac{1}{2}$ for $t \ge LT$, and

$$q(t) = \int_0^t g(\tau)\, d\tau \tag{4.9–5}$$

The signal pulse $g(t) = 0$ for $t < 0$ and $t \ge LT$. For $L = 1$, we have a full response CPM, and for $L > 1$, where L is a positive integer, we have a partial response CPM signal.

Now, when h is rational, i.e., $h = m/p$ where m and p are relatively prime positive integers, the CPM scheme can be represented by a trellis. In this case, there are *p phase states*

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \frac{2\pi m}{p}, \ldots, \frac{(p-1)\pi m}{p}\right\} \tag{4.9–6}$$

when m is even, and $2p$ phase states

$$\Theta_s = \left\{0, \frac{\pi m}{p}, \ldots, \frac{(2p-1)\pi m}{p}\right\} \tag{4.9–7}$$

when m is odd. If $L = 1$, these are the only states in the trellis. On the other hand, if $L > 1$, we have an additional number of states due to the partial response character of the signal pulse $g(t)$. These additional states can be identified by expressing $\theta(t; \boldsymbol{I})$ given by Equation 4.9–4 as

$$\theta(t; \boldsymbol{I}) = 2\pi h \sum_{k=n-L+1}^{n-1} I_k q(t - kT) + 2\pi h I_n q(t - nT) \tag{4.9–8}$$

The first term on the right-hand side of Equation 4.9–8 depends on the information symbols $(I_{n-1}, I_{n-2}, \ldots, I_{n-L+1})$, which is called the *correlative state vector*, and represents the phase term corresponding to signal pulses that have not reached their final value. The second term in Equation 4.9–8 represents the phase contribution due to the most recent symbol I_n. Hence, the state of the CPM signal (or the modulator) at time $t = nT$ may be expressed as the combined *phase state* and *correlative state*, denoted as

$$S_n = \{\theta_n, I_{n-1}, I_{n-2}, \ldots, I_{n-L+1}\} \tag{4.9–9}$$

for a partial response signal pulse of length LT, where $L > 1$. In this case, the number of states is

$$N_s = \begin{cases} pM^{L-1} & (\text{even } m) \\ 2pM^{L-1} & (\text{odd } m) \end{cases} \tag{4.9–10}$$

when $h = m/p$.

Now, suppose the state of the modulator at $t = nT$ is S_n. The effect of the new symbol in the time interval $nT \leq t \leq (n+1)T$ is to change the state from S_n to S_{n+1}. Hence, at $t = (n+1)T$, the state becomes

$$S_{n+1} = (\theta_{n+1}, I_n, I_{n-1}, \ldots, I_{n-L+2})$$

where

$$\theta_{n+1} = \theta_n + \pi h I_{n-L+1}$$

EXAMPLE 4.9-1. Consider a binary CPM scheme with a modulation index $h = 3/4$ and a partial response pulse with $L = 2$. Let us determine the states S_n of the CPM scheme and sketch the phase tree and state trellis.

First, we note that there are $2p = 8$ phase states, namely,

$$\Theta_s = \left\{0, \pm\tfrac{1}{4}\pi, \pm\tfrac{1}{2}\pi, \pm\tfrac{3}{4}\pi, \pi\right\}$$

For each of these phase states, there are two states that result from the memory of the CPM scheme. Hence, the total number of states is $N_s = 16$, namely,

$$\begin{aligned} &(0, 1), (0, -1), (\pi, 1), (\pi, -1), \left(\tfrac{1}{4}\pi, 1\right), \left(\tfrac{1}{4}\pi, -1\right), \left(\tfrac{1}{2}\pi, 1\right), \left(\tfrac{1}{2}\pi, -1\right), \\ &\left(\tfrac{3}{4}\pi, 1\right), \left(\tfrac{3}{4}\pi, -1\right), \left(-\tfrac{1}{4}\pi, 1\right), \left(-\tfrac{1}{4}\pi, -1\right), \left(-\tfrac{1}{2}\pi, 1\right), \left(-\tfrac{1}{2}\pi, -1\right), \\ &\left(-\tfrac{3}{4}\pi, 1\right), \left(-\tfrac{3}{4}\pi, -1\right) \end{aligned}$$

If the system is in phase state $\theta_n = -\frac{1}{4}\pi$ and $I_{n-1} = -1$, then

$$\begin{aligned} \theta_{n+1} &= \theta_n + \pi h I_{n-1} \\ &= -\tfrac{1}{4}\pi - \tfrac{3}{4}\pi = -\pi \end{aligned}$$

The state trellis is illustrated in Figure 4.9–1. A path through the state trellis corresponding to the sequence $(1, -1, -1, -1, 1, 1)$ is illustrated in Figure 4.9–2.

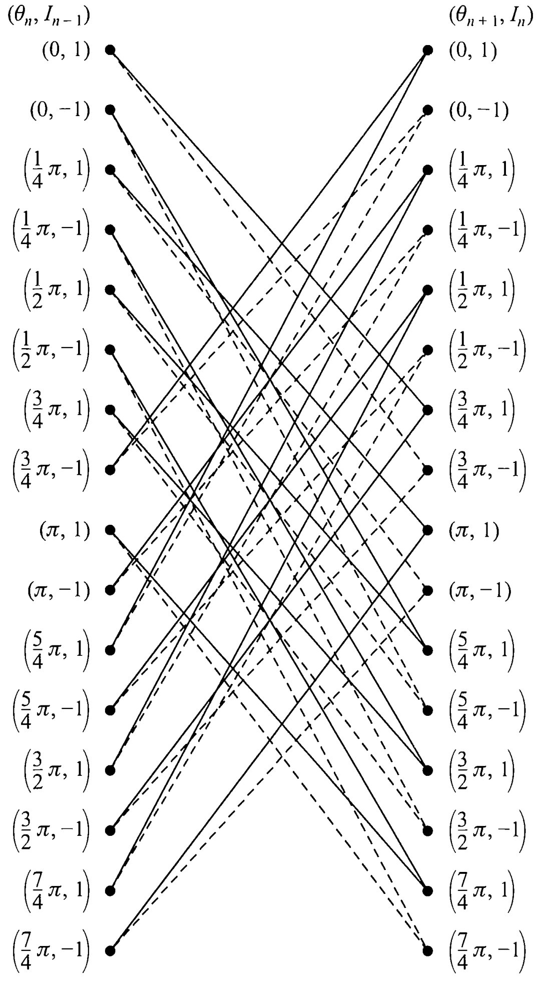

(θ_n, I_{n-1})
$(0, 1)$
$(0, -1)$
$(\frac{1}{4}\pi, 1)$
$(\frac{1}{4}\pi, -1)$
$(\frac{1}{2}\pi, 1)$
$(\frac{1}{2}\pi, -1)$
$(\frac{3}{4}\pi, 1)$
$(\frac{3}{4}\pi, -1)$
$(\pi, 1)$
$(\pi, -1)$
$(\frac{5}{4}\pi, 1)$
$(\frac{5}{4}\pi, -1)$
$(\frac{3}{2}\pi, 1)$
$(\frac{3}{2}\pi, -1)$
$(\frac{7}{4}\pi, 1)$
$(\frac{7}{4}\pi, -1)$
(θ_{n+1}, I_n)
$(0, 1)$
$(0, -1)$
$(\frac{1}{4}\pi, 1)$
$(\frac{1}{4}\pi, -1)$
$(\frac{1}{2}\pi, 1)$
$(\frac{1}{2}\pi, -1)$
$(\frac{3}{4}\pi, 1)$
$(\frac{3}{4}\pi, -1)$
$(\pi, 1)$
$(\pi, -1)$
$(\frac{5}{4}\pi, 1)$
$(\frac{5}{4}\pi, -1)$
$(\frac{3}{2}\pi, 1)$
$(\frac{3}{2}\pi, -1)$
$(\frac{7}{4}\pi, 1)$
$(\frac{7}{4}\pi, -1)$

FIGURE 4.9–1
State trellis for partial response ($L = 2$) CPM with $h = \frac{3}{4}$.

In order to sketch the phase tree, we must know the signal pulse shape $g(t)$. Figure 4.9–3 illustrates the phase tree when $g(t)$ is a rectangular pulse of duration $2T$, with initial state $(0, 1)$.

Having established the state trellis representation of CPM, let us now consider the metric computations performed in the Viterbi algorithm.

Metric Computations

By referring to the mathematical development for the derivation of the maximum likelihood demodulator given in Section 4.1, it is easy to show that the logarithm of the probability of the observed signal $r(t)$ conditioned on a particular sequence of transmitted symbols $\boldsymbol{I}$ is proportional to the cross-correlation metric

$$\begin{aligned} CM_n(\boldsymbol{I}) &= \int_{-\infty}^{(n+1)T} r(t)\cos[\omega_c t + \phi(t; \boldsymbol{I})]\,dt \\ &= CM_{n-1}(\boldsymbol{I}) + \int_{nT}^{(n+1)T} r(t)\cos[\omega_c t + \theta(t; \boldsymbol{I}) + \theta_n]\,dt \end{aligned} \tag{4.9–11}$$

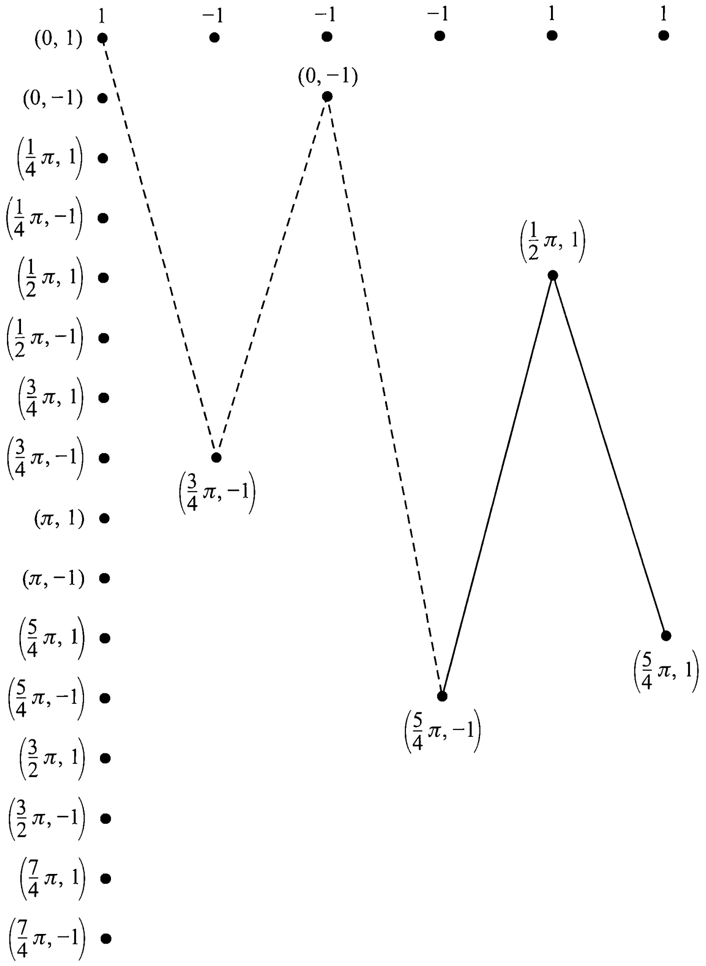

1
−1
−1
−1
1
1
(0, 1)
(0, −1)
$(\frac{1}{4}\pi, 1)$
$(\frac{1}{4}\pi, -1)$
$(\frac{1}{2}\pi, 1)$
$(\frac{1}{2}\pi, -1)$
$(\frac{3}{4}\pi, 1)$
$(\frac{3}{4}\pi, -1)$
$(\pi, 1)$
$(\pi, -1)$
$(\frac{5}{4}\pi, 1)$
$(\frac{5}{4}\pi, -1)$
$(\frac{3}{2}\pi, 1)$
$(\frac{3}{2}\pi, -1)$
$(\frac{7}{4}\pi, 1)$
$(\frac{7}{4}\pi, -1)$
(0, −1)
$(\frac{3}{4}\pi, -1)$
$(\frac{1}{2}\pi, 1)$
$(\frac{5}{4}\pi, -1)$
$(\frac{5}{4}\pi, 1)$

FIGURE 4.9–2
A single signal path through the trellis.

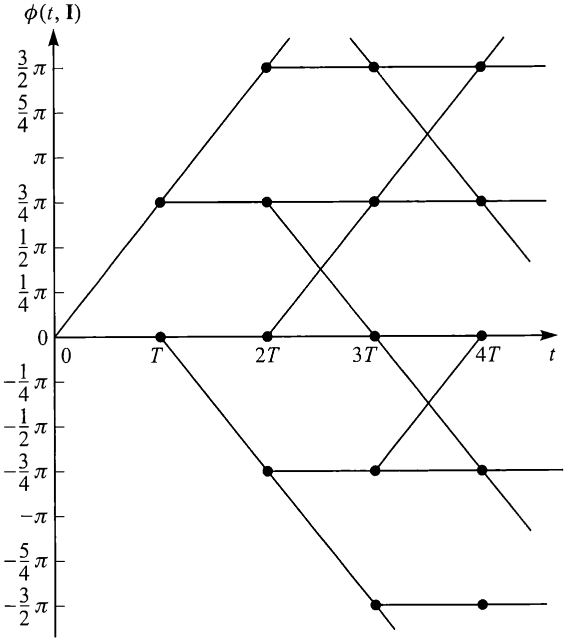

$\phi(t, \mathbf{I})$
$\frac{3}{2}\pi$
$\frac{5}{4}\pi$
π
$\frac{3}{4}\pi$
$\frac{1}{2}\pi$
$\frac{1}{4}\pi$
0
$-\frac{1}{4}\pi$
$-\frac{1}{2}\pi$
$-\frac{3}{4}\pi$
$-\pi$
$-\frac{5}{4}\pi$
$-\frac{3}{2}\pi$
0
T
$2T$
$3T$
$4T$
t

FIGURE 4.9–3
Phase tree for $L = 2$ partial response CPM with $h = \frac{3}{4}$.

The term $CM_{n-1}(\boldsymbol{I})$ represents the metrics for the surviving sequences up to time nT, and the term

$$v_n(\boldsymbol{I}; \theta_n) = \int_{nT}^{(n+1)T} r(t)\cos[\omega_c t + \theta(t; \boldsymbol{I}) + \theta_n]\,dt \tag{4.9–12}$$

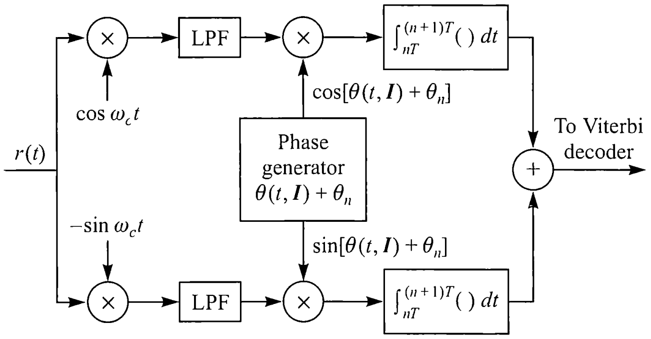
represents the additional increments to the metrics contributed by the signal in the time interval $nT \le t \le (n+1)T$. Note that there are M^L possible sequences $\boldsymbol{I} = (I_n, I_{n-1}, \ldots, I_{n-L+1})$ of symbols and p (or $2p$) possible phase states $\{\theta_n\}$. Therefore, there are pM^L (or $2pM^L$) different values of $v_n(\boldsymbol{I}, \theta_n)$ computed in each signal interval, and each value is used to increment the metrics corresponding to the pM^{L-1} surviving sequences from the previous signaling interval. A general block diagram that illustrates the computations of $v_n(\boldsymbol{I}; \theta_n)$ for the Viterbi decoder is shown in Figure 4.9–4.

Note that the number of surviving sequences at each state of the Viterbi decoding process is pM^{L-1} (or $2pM^{L-1}$). For each surviving sequence, we have M new increments of $v_n(\boldsymbol{I}; \theta_n)$ that are added to the existing metrics to yield pM^L (or $2pM^L$) sequences with pM^L (or $2pM^L$) metrics. However, this number is then reduced back to pM^{L-1} (or $2pM^{L-1}$) survivors with corresponding metrics by selecting the most probable sequence of the M sequences merging at each node of the trellis and discarding the other $M-1$ sequences.

4.9–2 Performance of CPM Signals

In evaluating the performance of CPM signals achieved with maximum-likelihood sequence detection, we must determine the minimum Euclidean distance of paths through the trellis that separate at the node at $t = 0$ and remerge at a later time at the same node. The distance between two paths through the trellis is related to the corresponding signals as we now demonstrate.

Suppose that we have two signals $s_i(t)$ and $s_j(t)$ corresponding to two phase trajectories $\phi(t; \boldsymbol{I}_i)$ and $\phi(t; \boldsymbol{I}_j)$. The sequences $\boldsymbol{I}_i$ and $\boldsymbol{I}_j$ must be different in their first symbol. Then, the Euclidean distance between the two signals over an interval of



×
LPF
×
$\int_{nT}^{(n+1)T}(\ )\,dt$
$\cos\omega_c t$
$\cos[\theta(t, I) + \theta_n]$
$r(t)$
Phase generator $\theta(t, I) + \theta_n$
To Viterbi decoder
+
$-\sin\omega_c t$
$\sin[\theta(t, I) + \theta_n]$
×
LPF
×
$\int_{nT}^{(n+1)T}(\ )\,dt$

FIGURE 4.9–4
Computation of metric increments $v_n(\boldsymbol{I}; \theta_n)$.

length NT, where $1/T$ is the symbol rate, is defined as

$$\begin{aligned}
d_{ij}^2 &= \int_0^{NT} [s_i(t) - s_j(t)]^2\, dt \\
&= \int_0^{NT} s_i^2(t)\, dt + \int_0^{NT} s_j^2(t)\, dt - 2\int_0^{NT} s_i(t)s_j(t)\, dt \\
&= 2N\mathcal{E} - 2\frac{2\mathcal{E}}{T}\int_0^{NT} \cos[\omega_c t + \phi(t; \boldsymbol{I}_i)]\cos[\omega_c t + \phi(t; \boldsymbol{I}_j)]\, dt \qquad (4.9\text{–}13) \\
&= 2N\mathcal{E} - \frac{2\mathcal{E}}{T}\int_0^{NT} \cos[\phi(t; \boldsymbol{I}_i) - \phi(t; \boldsymbol{I}_j)]\, dt \\
&= \frac{2\mathcal{E}}{T}\int_0^{NT} \{1 - \cos[\phi(t; \boldsymbol{I}_i) - \phi(t; \boldsymbol{I}_j)]\}\, dt
\end{aligned}$$

Hence the Euclidean distance is related to the phase difference between the paths in the state trellis according to Equation 4.9–13.

It is desirable to express the distance d_{ij}^2 in terms of the bit energy. Since $\mathcal{E} = \mathcal{E}_b \log_2 M$, Equation 4.9–13 may be expressed as

$$d_{ij}^2 = 2\mathcal{E}_b\delta_{ij}^2 \qquad (4.9\text{–}14)$$

where δ_{ij}^2 is defined as

$$\delta_{ij}^2 = \frac{\log_2 M}{T}\int_0^{NT} \{1 - \cos[\phi(t; \boldsymbol{I}_i) - \phi(t; \boldsymbol{I}_j)]\}\, dt \qquad (4.9\text{–}15)$$

Furthermore, we observe that $\phi(t; \boldsymbol{I}_i) - \phi(t; \boldsymbol{I}_j) = \phi(t; \boldsymbol{I}_i - \boldsymbol{I}_j)$, so that, with $\boldsymbol{\xi} = \boldsymbol{I}_i - \boldsymbol{I}_j$, Equation 4.9–15 may be written as

$$\delta_{ij}^2 = \frac{\log_2 M}{T}\int_0^{NT} [1 - \cos\phi(t; \boldsymbol{\xi})]\, dt \qquad (4.9\text{–}16)$$

where any element of $\boldsymbol{\xi}$ can take the values $0, \pm 2, \pm 4, \ldots, \pm 2(M-1)$, except that $\xi_0 \neq 0$.

The error rate performances for CPM is dominated by the term corresponding to the minimum Euclidean distance, and it may be expressed as

$$P_M = K_{\delta_{\min}} Q\left(\sqrt{\frac{\mathcal{E}_b}{N_0}\delta_{\min}^2}\right) \qquad (4.9\text{–}17)$$

where $K_{\delta_{\min}}$ is the number of paths having the minimum distance

$$\begin{aligned}
\delta_{\min}^2 &= \lim_{N\to\infty} \min_{i,j} \delta_{ij}^2 \\
&= \lim_{N\to\infty} \min_{i,j} \left\{\frac{\log_2 M}{T}\int_0^{NT} [1 - \cos\phi(t; \boldsymbol{I}_i - \boldsymbol{I}_j)]\, dt\right\}
\end{aligned} \qquad (4.9\text{–}18)$$

We note that for conventional binary PSK with no memory, $N = 1$ and $\delta_{\min}^2 = \delta_{12}^2 = 2$. Hence, Equation 4.9–17 agrees with our previous result.

Since $\delta_{\min}^2$ characterizes the performance of CPM, we can investigate the effect on $\delta_{\min}^2$ resulting from varying the alphabet size M, the modulation index h, and the length of the transmitted pulse in partial response CPM.

First, we consider full response ($L = 1$) CPM. If we take $M = 2$ as a beginning, we note that the sequences

$$\begin{aligned} \boldsymbol{I}_j &= +1, -1, I_2, I_3 \\ \boldsymbol{I}_j &= -1, +1, I_2, I_3 \end{aligned} \tag{4.9–19}$$

which differ for $k = 0, 1$ and agree for $k \geq 2$, result in two phase trajectories that merge after the second symbol. This corresponds to the difference sequence

$$\boldsymbol{\xi} = \{2, -2, 0, 0, \ldots\} \tag{4.9–20}$$

The Euclidean distance for this sequence is easily calculated from Equation 4.9–16, and provides an upper bound on $\delta^2_{\min}$. This upper bound for CPFSK with $M = 2$ is

$$d^2_B(h) = 2\left(1 - \frac{\sin 2\pi h}{2\pi h}\right), \qquad M = 2 \tag{4.9–21}$$

For example, where $h = \frac{1}{2}$, which corresponds to MSK, we have $d^2_B(\frac{1}{2}) = 2$, so that $\delta^2_{\min}\left(\frac{1}{2}\right) \leq 2$.

For $M > 2$ and full response CPM, it is also easily seen that phase trajectories merge at $t = 2T$. Hence, an upper bound on $\delta^2_{\min}$ can be obtained by considering the phase difference sequence $\boldsymbol{\xi} = \{\alpha, -\alpha, 0, 0, \ldots\}$ where $\alpha = \pm 2, \pm 4, \ldots, \pm 2(M-1)$. This sequence yields the upper bound for M-ary CPFSK as

$$d^2_B(h) = \min_{1 \leq k \leq M-1} \left\{(2 \log_2 M)\left(1 - \frac{\sin 2k\pi h}{2k\pi h}\right)\right\} \tag{4.9–22}$$

The graphs of $d^2_B(h)$ versus h for $M = 2, 4, 8, 16$ are shown in Figure 4.9–5. It is apparent from these graphs that large gains in performance can be achieved by increasing the alphabet size M. It must be remembered, however, that $\delta^2_{\min}(h) \leq d^2_B(h)$. That is, the upper bound may not be achievable for all values of h.

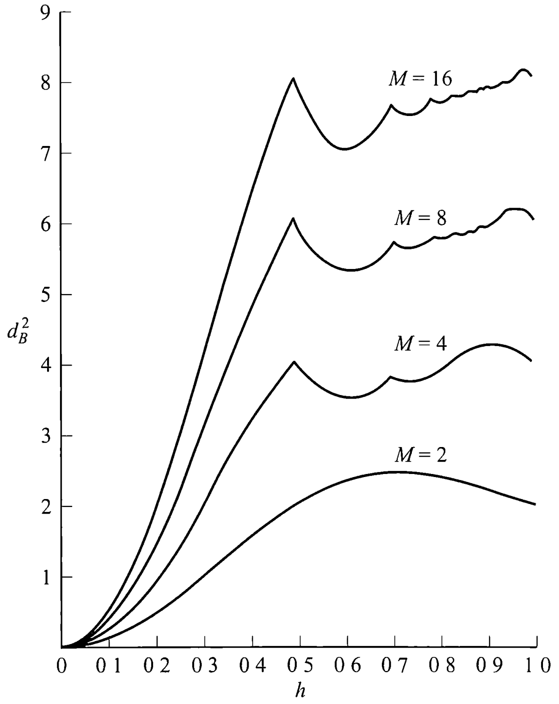


FIGURE 4.9–5
The upper bound d^2_B as a function of the modulation index h for full response CPM with rectangular pulses. [*From Aulin and Sundberg (1984). © 1984 John Wiley Ltd. Reprinted with permission of the publisher.*]

The minimum Euclidean distance $\delta^2_{\min}(h)$ has been determined, by evaluating Equation 4.9–16, for a variety of CPM signals by Aulin and Sundberg (1981). For example, Figure 4.9–6 illustrates the dependence of the Euclidean distance for binary CPFSK as a function of the modulation index h, with the number N of bit observation (decision) intervals ($N = 1, 2, 3, 4$) as a parameter. Also shown is the upper bound $d^2_B(h)$ given by Equation 4.9–21. In particular, we note that when $h = \frac{1}{2}$, $\delta^2_{\min}\left(\frac{1}{2}\right) = 2$, which is the same squared distance as PSK (binary or quaternary) with $N = 1$. On the other hand, the required observation interval for MSK is $N = 2$ intervals, for which we have $\delta^2_{\min}\left(\frac{1}{2}\right) = 2$. Hence, the performance of MSK with a Viterbi detector is comparable to (binary or quaternary) PSK as we have previously observed.

We also note from Figure 4.9–6 that the optimum modulation index for binary CPFSK is $h = 0.715$ when the observation interval is $N = 3$. This yields $\delta^2_{\min}(0.715) = 2.43$, or a gain of 0.85 dB relative to MSK.

Figure 4.9–7 illustrates the Euclidean distance as a function of h for $M = 4$ CPFSK, with the length of the observation interval N as a parameter. Also shown (as a dashed line where it is not reached) is the upper bound d^2_B evaluated from Equation 4.9–22. Note that $\delta^2_{\min}$ achieves the upper bound for several values of h for some N. In particular, note that the maximum value of d^2_B, which occurs at $h \approx 0.9$, is approximately reached for $N = 8$ observed symbol intervals. The true maximum is achieved at $h = 0.914$ with $N = 9$. For this case, $\delta^2_{\min}(0.914) = 4.2$, which represents a 3.2-dB gain over MSK. Also note that the Euclidean distance contains minima at $h = \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, 1$, etc. These values of h are called *weak modulation indices* and should be avoided. Similar results are available for larger values of M and may be found in the paper by Aulin and Sundberg (1981) and the text by Anderson et al. (1986).

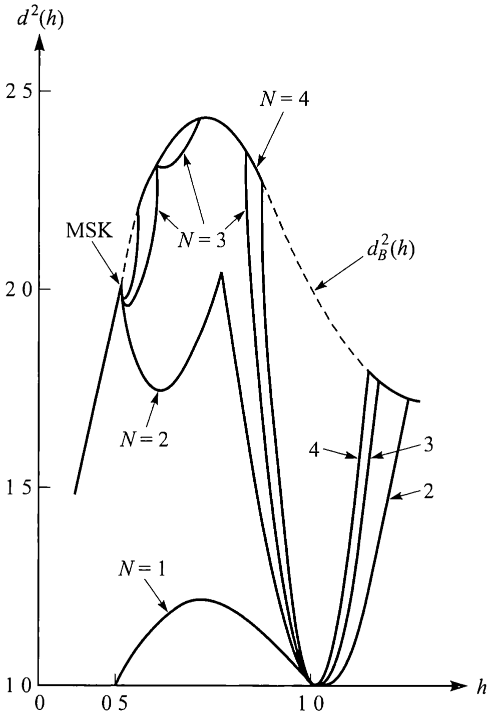


FIGURE 4.9–6

Squared minimum Euclidean distance as a function of the modulation index for binary CPFSK. The upper bound is d^2_B. [*From Aulin and Sundberg (1981), © 1981 IEEE.*]

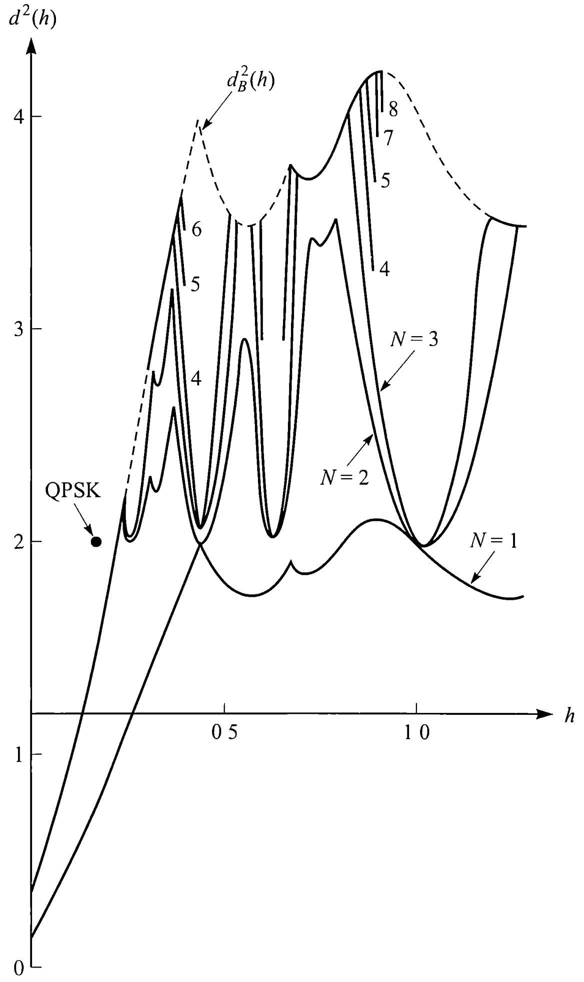


FIGURE 4.9–7
Squared minimum Euclidean distance as a function of the modulation index for quaternary CPFSK. The upper bound is d_B^2. [*From Aulin and Sundberg (1981), © 1981 IEEE.*]

Large performance gains can also be achieved with maximum-likelihood sequence detection of CPM by using partial response signals. For example, the distance bound $d_B^2(h)$ for partial response, raised cosine pulses given by

$$g(t) = \begin{cases} \dfrac{1}{2LT}\left(1 - \cos\dfrac{2\pi t}{2LT}\right) & 0 \le t \le LT \\ 0 & \text{otherwise} \end{cases} \tag{4.9–23}$$

is shown in Figure 4.9–8 for $M = 2$. Here, note that, as L increases, d_B^2 also achieves higher values. Clearly, the performance of CPM improves as the correlative memory L increases, but h must also be increased in order to achieve the larger values of d_B^2. Since a larger modulation index implies a larger bandwidth (for fixed L), while a larger memory length L (for fixed h) implies a smaller bandwidth, it is better to compare the Euclidean distance as a function of the normalized bandwidth $2WT_b$, where W is the 99 percent power bandwidth and T_b is the bit interval. Figure 4.9–9 illustrates this type of comparison with MSK used as a point of reference (0 dB). Note from this figure that there are several decibels to be gained by using partial response signals and higher signaling alphabets. The major price to be paid for this performance gain is the added exponentially increasing complexity in the implementation of the Viterbi detector.

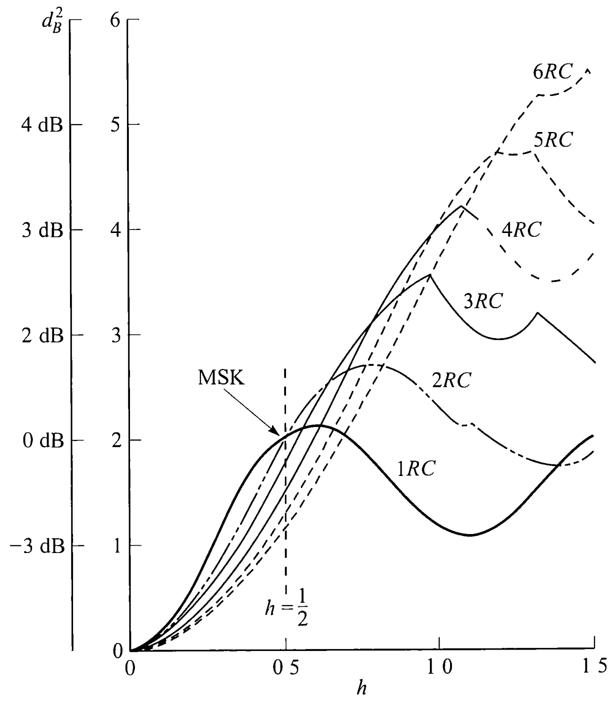


FIGURE 4.9–8

Upper bound d_B^2 on the minimum distance for partial response (raised cosine pulse) binary CPM. [*From Sundberg (1986), © 1986 IEEE.*]

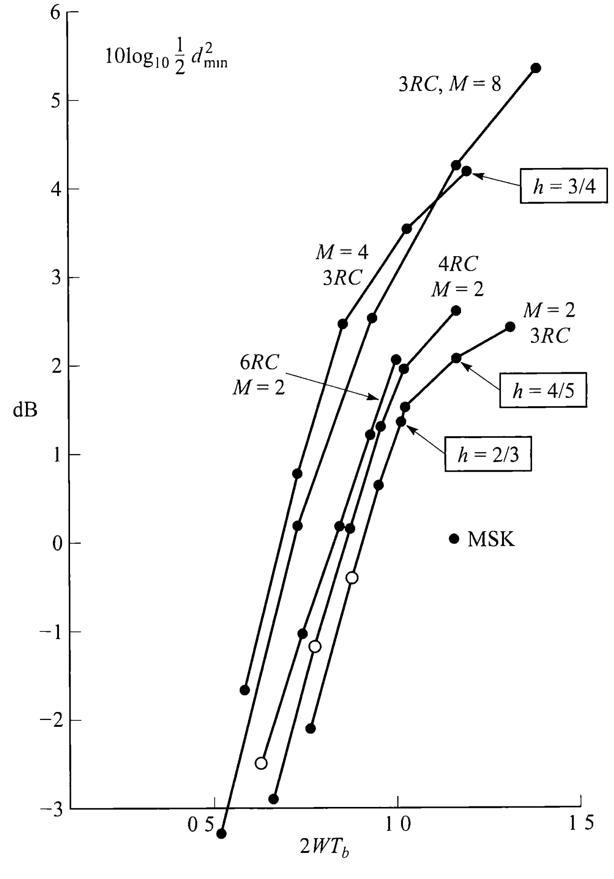


FIGURE 4.9–9

Power bandwidth tradeoff for partial response CPM signals with raised cosine pulses. *W* is the 99 percent inband power bandwidth. [*From Sundberg (1986), © 1986 IEEE.*]

The performance results shown in Figure 4.9–9 illustrate that a 3–4 dB gain relative to MSK can be easily obtained with relatively no increase in bandwidth by the use of raised cosine partial response CPM and $M = 4$. Although these results are for raised cosine signal pulses, similar gains can be achieved with other partial response pulse shapes. We emphasize that this gain in SNR is achieved by introducing memory into the signal modulation and exploiting the memory in the demodulation of the signal. No redundancy through coding has been introduced. In effect, the code has been built into the modulation and the trellis-type (Viterbi) decoding exploits the phase constraints in the CPM signal.

Additional gains in performance can be achieved by introducing additional redundancy through coding and increasing the alphabet size as a means of maintaining a fixed bandwidth. In particular, trellis-coded CPM using relatively simple convolution codes has been thoroughly investigated and many results are available in the technical literature. The Viterbi decoder for the convolutionally encoded CPM signal now exploits the memory inherent in the code and in the CPM signal. Performance gains of the order of 4–6 dB, relative to uncoded MSK with the same bandwidth, have been demonstrated by combining convolutional coding with CPM. Extensive numerical results for coded CPM are given by Lindell (1985).

Multi-*h* CPM

By varying the modulation index from one signaling interval to another, it is possible to increase the minimum Euclidean distance $\delta^2_{\min}$ between pairs of phase trajectories and, thus, improve the performance gain over constant-h CPM. Usually, multi-h CPM employs a fixed number H of modulation indices that are varied cyclically in successive signaling intervals. Thus, the phase of the signal varies piecewise linearly.

Significant gains in SNR are achievable by using only a small number of different values of h. For example, with full response ($L = 1$) CPM and $H = 2$, it is possible to obtain a gain of 3 dB relative to binary or quaternary PSK. By increasing H to $H = 4$, a gain of 4.5 dB relative to PSK can be obtained. The performance gain can also be increased with an increase in the signal alphabet. Table 4.9–1 lists the performance

TABLE 4.9–1
Maximum Values of the Upper Bound d_B^2 for Multi-h Linear Phase CPM[a]

M	H	Max d_B^2	dB gain compared with MSK	h_1	h_2	h_3	h_4	$\bar{h}$
2	1	2.43	0.85	0.715				0.715
2	2	4.0	3.0	0.5	0.5			0.5
2	3	4.88	3.87	0.620	0.686	0.714		0.673
2	4	5.69	4.54	0.73	0.55	0.73	0.55	0.64
4	1	4.23	3.25	0.914				0.914
4	2	6.54	5.15	0.772	0.772			0.772
4	3	7.65	5.83	0.795	0.795	0.795		0.795
8	1	6.14	4.87	0.964				0.964
8	2	7.50	5.74	0.883	0.883			0.883
8	3	8.40	6.23	0.879	0.879	0.879		0.879

[a] From Aulin and Sundberg (1982b)

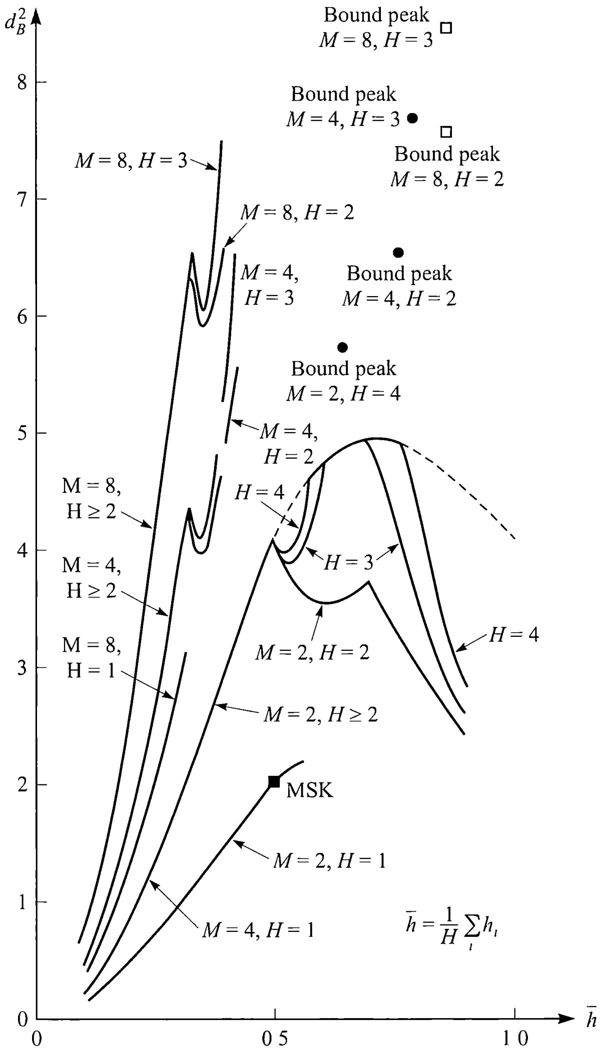


FIGURE 4.9–10

Upper bounds on minimum squared Euclidean distance for various M and H values. [*From Aulin and Sundberg (1982b), © 1982 IEEE.*]

gains achieved with $M = 2$, 4, and 8 for several values of H. The upper bounds on the minimum Euclidean distance are also shown in Figure 4.9–10 for several values of M and H. Note that the major gain in performance is obtained when H is increased from $H = 1$ to $H = 2$. For $H > 2$, the additional gain is relatively small for small values of $\{h_i\}$. On the other hand, significant performance gains are achieved by increasing the alphabet size M.

The results shown above hold for full response CPM. One can also extend the use of multi-h CPM to partial response in an attempt to further improve performance. It is anticipated that such schemes will yield some additional performance gains, but numerical results on partial response, multi-h CPM are limited. The interested reader is referred to the paper by Aulin and Sundberg (1982b).

4.9–3 Suboptimum Demodulation and Detection of CPM Signals

The high complexity inherent in the implementation of the maximum-likelihood sequence detector for CPM signals has been a motivating factor in the investigation of reduced-complexity detectors. Reduced-complexity Viterbi detectors were investigated by Svensson (1984), Svensson et al. (1984), Svensson and Sundberg (1983), Aulin et al. (1981), Simmons and Wittke (1983), Palenius and Svensson (1993), and Palenius (1991). The basic idea in achieving a reduced-complexity Viterbi detector is to design a receiver filter that has a shorter pulse than the transmitter. The receiver pulse $g_R(t)$ must be chosen in such a way that the phase tree generated by $g_R(t)$ is a good approximation of the phase tree generated by the transmitter pulse $g_T(t)$. Performance results indicate that a significant reduction in complexity can be achieved at a loss in performance of about 0.5 to 1 dB.

Another method for reducing the complexity of the receiver for CPM signals is to exploit the linear representation of CPM, which can be expressed as a sum of amplitude-modulated pulses as given in the papers by Laurent (1986) and Mengali and Morelli (1995). In many cases of practical interest the CPM signal can be approximated by a single amplitude-modulated pulse or, perhaps, by a sum of two amplitude-modulated pulses. Hence, the receiver can be easily implemented based on this linear representation of the CPM signal. The performance of such relatively simple receivers has been investigated by Kawas-Kaleh (1989). The results of this study indicate that such simplified receivers sacrifice little in performance but achieve a significant reduction in implementation complexity.

4.10

PERFORMANCE ANALYSIS FOR WIRELINE AND RADIO COMMUNICATION SYSTEMS

In the transmission of digital signals through an AWGN channel, we have observed that the performance of the communication system, measured in terms of the probability of error, depends solely on the received SNR, $\mathcal{E}_b/N_0$, where $\mathcal{E}_b$ is the transmitted energy per bit and $\frac{1}{2}N_0$ is the power spectral density of the additive noise. Hence, the additive noise ultimately limits the performance of the communication system.

In addition to the additive noise, another factor that affects the performance of a communication system is channel attenuation. All physical channels, including wire lines and radio channels, are lossy. Hence, the signal is attenuated as it travels through the channel. The simple mathematical model for the attenuation shown in Figure 4.10–1 may be used for the channel. Consequently, if the transmitted signal is $s(t)$, the received signal, with $0 < \alpha \leq 1$ is

$$r(t) = \alpha s(t) + n(t) \tag{4.10–1}$$

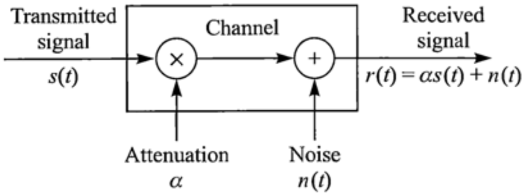


FIGURE 4.10–1
Mathematical model of channel with attenuation and additive noise.

Then, if the energy in the transmitted signal is $\mathcal{E}_b$, the energy in the received signal is $\alpha^2\mathcal{E}_b$. Consequently, the received signal has an SNR $\alpha^2\mathcal{E}_b/N_0$. Hence, the effect of signal attenuation is to reduce the energy in the received signal and thus to render the communication system more vulnerable to additive noise.

In analog communication systems, amplifiers called repeaters are used to periodically boost the signal strength in transmission through the channel. However, each amplifier also boosts the noise in the system. In contrast, digital communication systems allow us to detect and regenerate a clean (noise-free) signal in a transmission channel. Such devices, called *regenerative repeaters*, are frequently used in wireline and fiber-optic communication channels.

4.10–1 Regenerative Repeaters

The front end of each regenerative repeater consists of a demodulator/detector that demodulates and detects the transmitted digital information sequence sent by the preceding repeater. Once detected, the sequence is passed to the transmitter side of the repeater, which maps the sequence into signal waveforms that are transmitted to the next repeater. This type of repeater is called a regenerative repeater.

Since a noise-free signal is regenerated at each repeater, the additive noise does not accumulate. However, when errors occur in the detector of a repeater, the errors are propagated forward to the following repeaters in the channel. To evaluate the effect of errors on the performance of the overall system, suppose that the modulation is binary PAM, so that the probability of a bit error for one hop (signal transmission from one repeater to the next repeater in the chain) is

$$P_b = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

Since errors occur with low probability, we may ignore the probability that any one bit will be detected incorrectly more than once in transmission through a channel with K repeaters. Consequently, the number of errors will increase linearly with the number of regenerative repeaters in the channel, and therefore, the overall probability of error may be approximated as

$$P_b \approx KQ\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right) \tag{4.10–2}$$

In contrast, the use of K analog repeaters in the channel reduces the received SNR by K, and hence, the bit-error probability is

$$P_b \approx Q\left(\sqrt{\frac{2\mathcal{E}_b}{KN_0}}\right) \tag{4.10–3}$$

Clearly, for the same probability of error performance, the use of regenerative repeaters results in a significant saving in transmitter power compared with analog repeaters.

Hence, in digital communication systems, regenerative repeaters are preferable. However, in wireline telephone channels that are used to transmit both analog and digital signals, analog repeaters are generally employed.

EXAMPLE 4.10–1. A binary digital communication system transmits data over a wireline channel of length 1000 km. Repeaters are used every 10 km to offset the effect of channel attenuation. Let us determine the $\mathcal{E}_b/N_0$ that is required to achieve a probability of a bit error of 10^{-5} if (a) analog repeaters are employed, and (b) regenerative repeaters are employed.

The number of repeaters used in the system is $K = 100$. If regenerative repeaters are used, the $\mathcal{E}_b/N_0$ obtained from Equation 4.10–2 is

$$10^{-5} = 100Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$
$$10^{-7} = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}}\right)$$

which yields approximately 11.3 dB. If analog repeaters are used, the $\mathcal{E}_b/N_0$ obtained from Equation 4.10–3 is

$$10^{-5} = Q\left(\sqrt{\frac{2\mathcal{E}_b}{100N_0}}\right)$$

which yields $\mathcal{E}_b/N_0 \approx 29.6$ dB. Hence, the difference in the required SNR is about 18.3 dB, or approximately 70 times the transmitter power of the digital communication system.

4.10–2 Link Budget Analysis in Radio Communication Systems

In the design of radio communication systems that transmit over line-of-sight microwave channels and satellite channels, the system designer must specify the size of the transmit and receive antennas, the transmitted power, and the SNR required to achieve a given level of performance at some desired data rate. The system design procedure is relatively straightforward and is outlined below.

Let us begin with a transmit antenna that radiates isotropically in free space at a power level of P_T watts as shown in Figure 4.10–2. The power density at a distance d from the antenna is $P_T/4\pi d^2$ W/m^2. If the transmitting antenna has some directivity in

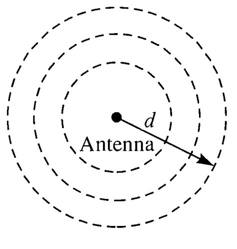


FIGURE 4.10–2
Isotropically radiating antenna.

a particular direction, the power density in that direction is increased by a factor called the antenna gain and denoted by G_T. In such a case, the power density at distance d is $P_T G_T / 4\pi d^2$ W/m^2. The product $P_T G_T$ is usually called the *effective radiated power* (ERP or EIRP), which is basically the radiated power relative to an isotropic antenna, for which $G_T = 1$.

A receiving antenna pointed in the direction of the radiated power gathers a portion of the power that is proportional to its cross-sectional area. Hence, the received power extracted by the antenna may be expressed as

$$P_R = \frac{P_T G_T A_R}{4\pi d^2} \tag{4.10–4}$$

where A_R is the *effective area of the antenna*. From electromagnetic field theory, we obtain the basic relationship between the gain G_R of an antenna and its effective area as

$$A_R = \frac{G_R \lambda^2}{4\pi} \quad \mathrm{m}^2 \tag{4.10–5}$$

where $\lambda = c/f$ is the wavelength of the transmitted signal, c is the speed of light (3×10^8 m/s), and f is the frequency of the transmitted signal.

If we substitute Equation 4.10–5 for A_R into Equation 4.10–4, we obtain an expression for the received power in the form

$$P_R = \frac{P_T G_T G_R}{(4\pi d/\lambda)^2} \tag{4.10–6}$$

The factor

$$L_s = \left(\frac{\lambda}{4\pi d}\right)^2 \tag{4.10–7}$$

is called the *free-space path loss*. If other losses, such as atmospheric losses, are encountered in the transmission of the signal, they may be accounted for by introducing an additional loss factor, say L_a. Therefore, the received power may be written in general as

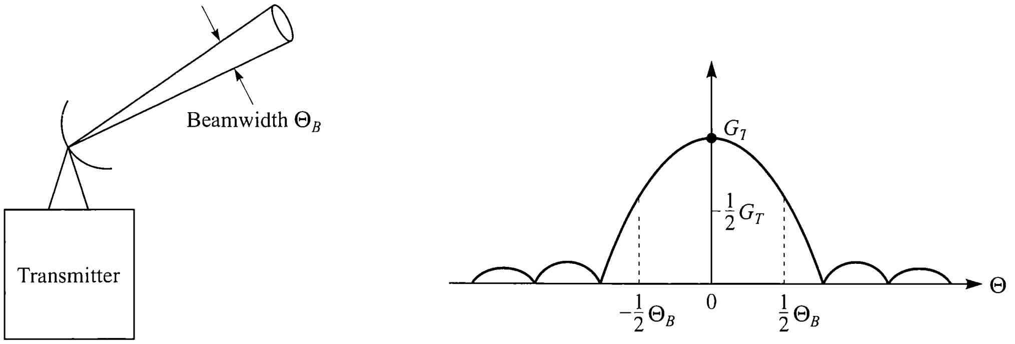
$$P_R = P_T G_T G_R L_s L_a \tag{4.10–8}$$

As indicated above, the important characteristics of an antenna are its gain and its effective area. These generally depend on the wavelength of the radiated power and the physical dimensions of the antenna. For example, a parabolic (dish) antenna of diameter D has an effective area

$$A_R = \tfrac{1}{4}\pi D^2 \eta \tag{4.10–9}$$

where $\frac{1}{4}\pi D^2$ is the physical area and η is the *illumination efficiency factor*, which falls in the range $0.5 \le \eta \le 0.6$. Hence, the antenna gain for a parabolic antenna of diameter D is

$$G_R = \eta \left(\frac{\pi D}{\lambda}\right)^2 \tag{4.10–10}$$



(a) Beamwidth of antenna

(b) Antenna pattern

FIGURE 4.10–3
Antenna beamwidth and pattern.

As a second example, a horn antenna of physical area A has an efficiency factor of 0.8, an effective area of $A_R = 0.8A$, and an antenna gain of

$$G_R = \frac{10A}{\lambda^2} \tag{4.10–11}$$

Another parameter that is related to the gain (directivity) of an antenna is its beamwidth, which we denote as Θ_B and which is illustrated graphically in Figure 4.10–3. Usually, the beamwidth is measured as the -3 dB width of the antenna pattern. For example, the -3 dB beamwidth of a parabolic antenna is approximately

$$\Theta_B = 70(\lambda/D)^\circ \tag{4.10–12}$$

so that G_T is inversely proportional to Θ_B^2. That is, a decrease of the beamwidth by a factor of 2, which is obtained by doubling the diameter D, increases the antenna gain by a factor of 4 (6 dB).

Based on the general relationship for the received signal power given by Equation 4.10–8, the system designer can compute P_R from a specification of the antenna gains and the distance between the transmitter and the receiver. Such computations are usually done on a power basis, so that

$$(P_R)_{\text{dB}} = (P_T)_{\text{dB}} + (G_T)_{\text{dB}} + (G_R)_{\text{dB}} + (L_s)_{\text{dB}} + (L_a)_{\text{dB}} \tag{4.10–13}$$

EXAMPLE 4.10–2. Suppose that we have a satellite in geosynchronous orbit (36,000 km above the earth's surface) that radiates 100 W of power, i.e., 20 dB above 1 W (20 dBW). The transmit antenna has a gain of 17 dB, so that the ERP = 37 dBW. Also, suppose that the earth station employs a 3-m parabolic antenna and that the downlink is operating at a frequency of 4 GHz. The efficiency factor is $\eta = 0.5$. By substituting these numbers into Equation 4.10–10, we obtain the value of the antenna gain as 39 dB. The free-space path loss is

$$L_s = 195.6\,\text{dB}$$

No other losses are assumed. Therefore, the received signal power is

$$\begin{aligned}(P_R)_{\text{dB}} &= 20 + 17 + 39 - 195.6 \\ &= -119.6\,\text{dBW}\end{aligned}$$

or, equivalently,

$$P_R = 1.1 \times 10^{-12}\,\mathrm{W}$$

To complete the link budget computation, we must consider the effect of the additive noise at the receiver front end. Thermal noise that arises at the receiver front end has a relatively flat power density spectrum up to about 10^{12} Hz, and is given as

$$N_0 = k_B T_0 \quad \mathrm{W/Hz} \tag{4.10–14}$$

where k_B is Boltzmann's constant (1.38×10^{-23} W-s/K) and T_0 is the noise temperature in Kelvin. Therefore, the total noise power in the signal bandwidth W is $N_0 W$.

The performance of the digital communication system is specified by the $\mathcal{E}_b/N_0$ required to keep the error rate performance below some given value. Since

$$\frac{\mathcal{E}_b}{N_0} = \frac{T_b P_R}{N_0} = \frac{1}{R}\frac{P_R}{N_0} \tag{4.10–15}$$

it follows that

$$\frac{P_R}{N_0} = R\left(\frac{\mathcal{E}_b}{N_0}\right)_{\mathrm{req}} \tag{4.10–16}$$

where $(\mathcal{E}_b/N_0)_{\mathrm{req}}$ is the required SNR per bit. Hence, if we have P_R/N_0 and the required SNR per bit, we can determine the maximum data rate that is possible.

EXAMPLE 4.10–3. For the link considered in Example 4.10–2, the received signal power is

$$P_R = 1.1 \times 10^{-12}\,\mathrm{W} \quad (-119.6\ \mathrm{dBW})$$

Now, suppose the receiver front end has a noise temperature of 300 K, which is typical for a receiver in the 4-GHz range. Then

$$N_0 = 4.1 \times 10^{-21}\,\mathrm{W/Hz}$$

or, equivalently, −203.9 dBW/Hz. Therefore,

$$\frac{P_R}{N_0} = -119.6 + 203.9 = 84.3\ \mathrm{dB\ Hz}$$

If the required SNR per bit is 10 dB, then, from Equation 4.10–16, we have the available rate as

$$\begin{aligned} R_{\mathrm{dB}} &= 84.3 - 10 \\ &= 74.3\ \mathrm{dB} \qquad (\text{with respect to 1 bit/s}) \end{aligned}$$

This corresponds to a rate of 26.9 megabits/s, which is equivalent to about 420 PCM channels, each operating at 64,000 bits/s.

It is a good idea to introduce some safety margin, which we shall call the *link margin* M_{dB}, in the above computations for the capacity of the communication link. Typically, this may be selected as $M_{\mathrm{dB}} = 6$ dB. Then, the link budget computation for the link capacity may be expressed in the simple form

$$\begin{aligned} R_{\mathrm{dB}} &= \left(\frac{P_R}{N_0}\right)_{\mathrm{dB\ Hz}} - \left(\frac{\mathcal{E}_b}{N_0}\right)_{\mathrm{req}} - M_{\mathrm{dB}} \\ &= (P_T)_{\mathrm{dBW}} + (G_T)_{\mathrm{dB}} + (G_R)_{\mathrm{dB}} \\ &\quad + (L_a)_{\mathrm{dB}} + (L_s)_{\mathrm{dB}} - (N_0)_{\mathrm{dBW/Hz}} - \left(\frac{\mathcal{E}_b}{N_0}\right)_{\mathrm{req}} - M_{\mathrm{dB}} \end{aligned} \tag{4.10–17}$$

4.11
BIBLIOGRAPHICAL NOTES AND REFERENCES

In the derivation of the optimum demodulator for a signal corrupted by AWGN, we applied mathematical techniques that were originally used in deriving optimum receiver structures for radar signals. For example, the matched filter was first proposed by North (1943) for use in radar detection, and it is sometimes called the North filter. An alternative method for deriving the optimum demodulator and detector is the Karhunen–Loeve expansion, which is described in the classical texts by Davenport and Root (1958), Helstrom (1968), and Van Trees (1968). Its use in radar detection theory is described in the paper by Kelly et al. (1960). These detection methods are based on the hypothesis testing methods developed by statisticians, e.g., Neyman and Pearson (1933) and Wald (1947).

The geometric approach to signal design and detection, which was presented in the context of digital modulation and which has its roots in Kotelnikov (1947) and Shannon's original work, is conceptually appealing and is now widely used since its use in the text by Wozencraft and Jacobs (1965).

Design and analysis of signal constellations for the AWGN channel have received considerable attention in the technical literature. Of particular significance is the performance analysis of two-dimensional (QAM) signal constellations that has been treated in the papers of Cahn (1960), Hancock and Lucky (1960), Campopiano and Glazer (1962), Lucky and Hancock (1962), Salz et al. (1971), Simon and Smith (1973), Thomas et al. (1974), and Foschini et al. (1974). Signal design based on multidimensional signal constellations has been described and analyzed in the paper by Gersho and Lawrence (1984).

The Viterbi algorithm was devised by Viterbi (1967) for the purpose of decoding convolutional codes. Its use as the optimal maximum-likelihood sequence detection algorithm for signals with memory was described by Forney (1972) and Omura (1971). Its use for carrier modulated signals was considered by Ungerboeck (1974) and MacKenchnie (1973). It was subsequently applied to the demodulation of CPM by Aulin and Sundberg (1981), Aulin et al. (1981), and Aulin (1980).

Our discussion of the demodulation and detection of signals with memory referenced journal papers published primarily in the United States. The authors have recently learned that maximum-likelihood sequential detection algorithms for signals with memory (introduced by the channel through intersymbol interference) were also developed and published in Russia during the 1960s by D. Klovsky. An English translation of Klovsky's work is contained in his book coauthored with B. Nikolaev (1978).

PROBLEMS

4.1 Let $Z(t) = X(t) + jY(t)$ be a complex-valued, zero-mean white Gaussian noise process with autocorrelation function $R_Z(\tau) = N_0\delta(\tau)$. Let $f_m(t),\ m = 1, 2, \ldots, M$, be a set of M orthogonal equivalent lowpass waveforms defined on the interval $0 \le t \le T$. Define

$$N_{mr} = \text{Re}\left[\int_0^T Z(t) f_m^*(t)\, dt\right], \qquad m = 1, 2, \ldots, M$$

1. Determine the variance of N_{mr}.
2. Show that $\text{E}[N_{mr}N_{kr}] = 0$ for $k \neq m$.

4.2 The correlation metrics given by Equation 4.2–28 are

$$C(\boldsymbol{r}, \boldsymbol{s}_m) = 2\sum_{n=1}^{N} r_n s_{mn} - \sum_{n=1}^{N} s_{mn}^2, \qquad m = 1, 2, \ldots, M$$

where

$$r_n = \int_0^T r(t)\,\phi_n(t)\, dt$$

and

$$s_{mn} = \int_0^T s_m(t)\,\phi_n(t)\, dt$$

Show that the correlation metrics are equivalent to the metrics

$$C(\boldsymbol{r}, \boldsymbol{s}_m) = 2\int_0^T r(t)\, s_m(t)\, dt - \int_0^T s_m^2(t)\, dt$$

4.3 In the communication system shown in Figure P4.3, the receiver receives two signals r_1 and r_2, where r_2 is a "noisier" version of r_1. The two noises n_1 and n_2 are arbitrary—not necessarily Gaussian, and not necessarily independent. Intuition would suggest that since r_2 is noisier than r_1, the optimal decision can be based only on r_1; in other words, r_2 is irrelevant. Is this true or false? If it is true, give a proof; if it is false, provide a counterexample and state under what conditions this can be true.

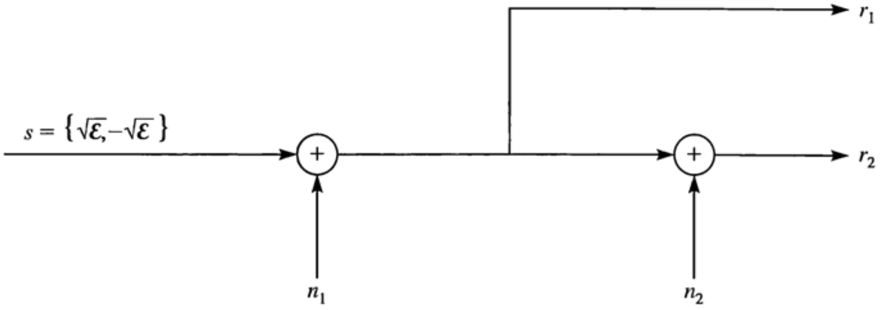


FIGURE P4.3

4.4 A binary digital communication system employs the signals

$$s_0(t) = 0 \qquad 0 \le t \le T$$
$$s_1(t) = A \qquad 0 \le t \le T$$

for transmitting the information. This is called *on-off signaling*. The demodulator cross-correlates the received signal $r(t)$ with $s(t)$ and samples the output of the correlator at $t + T$.

a. Determine the optimum detector for an AWGN channel and the optimum threshold, assuming that the signals are equally probable.

b. Determine the probability of error as a function of the SNR. How does on-off signaling compare with antipodal signaling?

4.5 A communication system transmits one of the three messages m_1, m_2, and m_3 using signals $s_1(t)$, $s_2(t)$, and $s_3(t)$. The signal $s_3(t) = 0$, and $s_1(t)$ and $s_2(t)$ are shown in Figure P4.5. The channel is an additive white Gaussian noise channel with noise power spectral density equal to $N_0/2$.

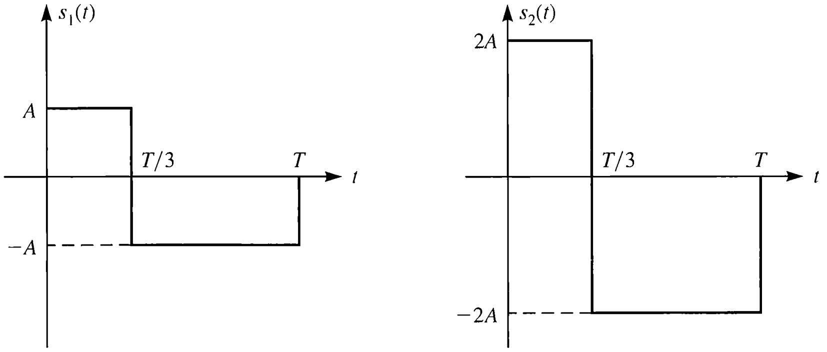


FIGURE P4.5

1. Determine an orthonormal basis for this signal set, and depict the signal constellation.
2. If the three messages are equiprobable, what are the optimal decision rules for this system? Show the optimal decision regions on the signal constellation you plotted in part 1.
3. If the signals are equiprobable, express the error probability of the optimal detector in terms of the average SNR per bit.
4. Assuming this system transmits 3000 symbols per second, what is the resulting transmission rate (in bits per second)?

4.6 Suppose that binary PSK is used for transmitting information over an AWGN with a power spectral density of $\frac{1}{2}N_0 = 10^{-10}$ W/Hz. The transmitted signal energy is $\mathcal{E}_b = \frac{1}{2}A^2T$, where T is the bit interval and A is the signal amplitude. Determine the signal amplitude required to achieve an error probability of 10^{-6} when the data rate is

1. 10 kilobits/s
2. 100 kilobits/s
3. 1 megabit/s

4.7 Consider a signal detector with an input

$$r = \pm A + n$$

where $+A$ and $-A$ occur with equal probability and the noise variable n is characterized by the (Laplacian) PDF shown in Figure P4.7.

1. Determine the probability of error as a function of the parameters A and σ.
2. Determine the SNR required to achieve an error probability of 10^{-5}. How does the SNR compare with the result for a Gaussian PDF?

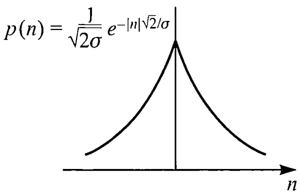


FIGURE P4.7

4.8 The signal constellation for a communication system with 16 equiprobable symbols is shown in Figure P4.8. The channel is AWGN with noise power spectral density of $N_0/2$.

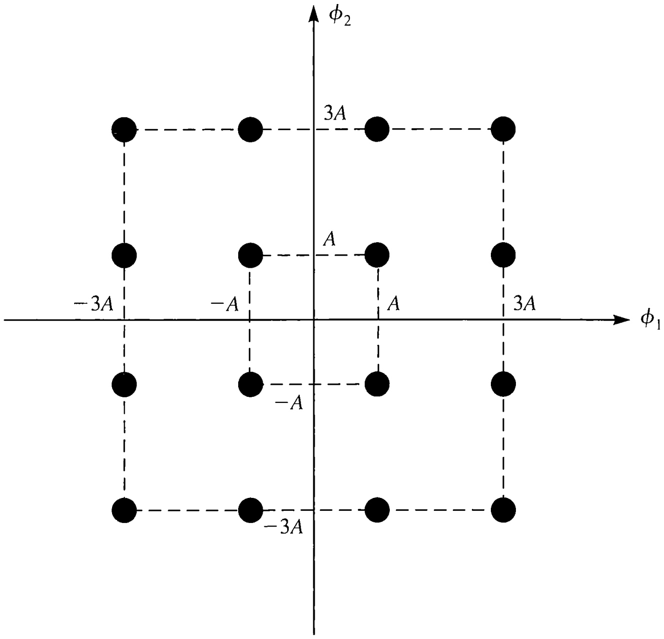


FIGURE P4.8

1. Using the union bound, find a bound in terms of A and N_0 on the error probability for this channel.
2. Determine the average SNR per bit for this channel.
3. Express the bound found in part 1 in terms of the average SNR per bit.
4. Compare the power efficiency of this system with a 16-level PAM system.

4.9 A ternary communication system transmits one of three equiprobable signals $s(t)$, 0, or $-s(t)$ every T seconds. The received signal is $r_l(t) = s(t) + z(t)$, $r_l(t) = z(t)$, or $r_l(t) = -s(t) + z(t)$, where $z(t)$ is white Gaussian noise with $\mathrm{E}[z(t)] = 0$ and $R_z(\tau) = \mathrm{E}[z(t)z^*(\tau)] = 2N_0\delta(t - \tau)$. The optimum receiver computes the correlation metric

$$U = \mathrm{Re}\left[\int_0^T r_l(t)s^*(t)\,dt\right]$$

and compares U with a threshold A and a threshold $-A$. If $U > A$, the decision is made that $s(t)$ was sent. If $U < -A$, the decision is made in favor of $-s(t)$. If $-A < U < A$, the decision is made in favor of 0.

1. Determine the three conditional probabilities of error: P_e given that $s(t)$ was sent, P_e given that $-s(t)$ was sent, and P_e given that 0 was sent.
2. Determine the average probability of error P_e as a function of the threshold A, assuming that the three symbols are equally probable a priori.
3. Determine the value of A that minimizes P_e.

4.10 The two equivalent lowpass signals shown in Figure P4.10 are used to transmit a binary information sequence. The transmitted signals, which are equally probable, are corrupted by additive zero-mean white Gaussian noise having an equivalent lowpass representation $z(t)$ with an autocorrelation function

$$R_Z(\tau) = \mathrm{E}\left[z^*(t)\,z(t+\tau)\right] = 2N_0\delta(\tau)$$

1. What is the transmitted signal energy?
2. What is the probability of a binary digit error if coherent detection is employed at the receiver?
3. What is the probability of a binary digit error if noncoherent detection is employed at the receiver?

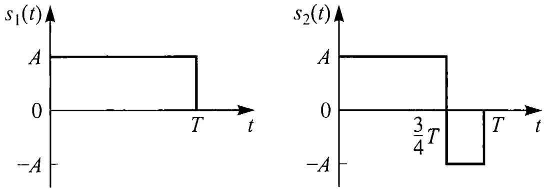


FIGURE P4.10

4.11 A matched filter has the frequency response

$$H(f) = \frac{1 - e^{-j2\pi fT}}{j2\pi f}$$

1. Determine the impulse response $h(t)$ corresponding to $H(f)$.
2. Determine the signal waveform to which the filter characteristic is matched.

4.12 Consider the signal

$$s(t) = \begin{cases} (A/T)t\cos 2\pi f_c t & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases}$$

1. Determine the impulse response of the matched filter for the signal.
2. Determine the output of the matched filter at $t = T$.
3. Suppose the signal $s(t)$ is passed through a correlator that correlates the input $s(t)$ with $s(t)$. Determine the value of the correlator output at $t = T$. Compare your result with that in part 2.

4.13 The two equivalent lowpass signals shown in Figure P4.13 are used to transmit a binary sequence over an additive white Gaussian noise channel. The received signal can be expressed as

$$r_i(t) = s_i(t) + z(t), \qquad 0 \le t \le T, \qquad i = 1, 2$$

where $z(t)$ is a zero-mean Gaussian noise process with autocorrelation function

$$R_Z(\tau) = \mathrm{E}\left[z^*(t)z(t+\tau)\right] = 2N_0\delta(\tau)$$

1. Determine the transmitted energy in $s_1(t)$ and $s_2(t)$ and the cross-correlation coefficient ρ_{12}.
2. Suppose the receiver is implemented by means of coherent detection using two matched filters, one matched to $s_1(t)$ and the other to $s_2(t)$. Sketch the equivalent lowpass impulse responses of the matched filters.

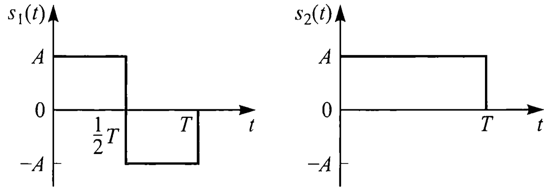


FIGURE P4.13

3. Sketch the noise-free response of the two matched filters when the transmitted signal is $s_2(t)$.
4. Suppose the receiver is implemented by means of two cross-correlators (multipliers followed by integrators) in parallel. Sketch the output of each integrator as a function of time for the interval $0 \leq t \leq T$ when the transmitted signal is $s_2(t)$.
5. Compare the sketches in parts 3 and 4. Are they the same? Explain briefly.
6. From your knowledge of the signal characteristics, give the probability of error for this binary communication system.

4.14 A binary communication system uses two equiprobable messages $s_1(t) = p(t)$ and $s_2(t) = -p(t)$. The channel noise is additive white Gaussian with power spectral density $N_0/2$. Assume that we have designed an optimal receiver for this channel, and let the error probability for the optimal receiver be P_e.

1. Find an expression for P_e.
2. If this receiver is used on an AWGN channel using the same signals but with the noise power spectral density $N_1 > N_0$, find the resulting error probability P_1 and explain how its value compares with P_e.
3. Let P_{e1} denote the error probability in part 2 *when an optimal receiver is designed for the new noise power spectral density* N_1. Find P_{e1} and compare it with P_1.
4. Answer parts 1 and 2 if the two signals are not equiprobable but have prior probabilities p and $1 - p$.

4.15 Consider a quaternary ($M = 4$) communication system that transmits, every T seconds, one of four equally probable signals: $s_1(t)$, $-s_1(t)$, $s_2(t)$, $-s_2(t)$. The signals $s_1(t)$ and $s_2(t)$ are orthogonal with equal energy. The additive noise is white Gaussian with zero mean and autocorrelation function $R_z(\tau) = N_0/2\delta(\tau)$. The demodulator consists of two filters matched to $s_1(t)$ and $s_2(t)$, and their outputs at the sampling instant are U_1 and U_2. The detector bases its decision on the following rule:

$$U_1 > |U_2| \Rightarrow s_1(t) \qquad U_1 < -|U_2| \Rightarrow -s_1(t)$$
$$U_2 > |U_1| \Rightarrow s_2(t) \qquad U_2 < -|U_1| \Rightarrow -s_2(t)$$

Since the signal set is biorthogonal, the error probability is given by $(1 - P_c)$, where P_c is given by Equation 4.4–26. Express this error probability in terms of a single integral, and thus show that the symbol error probability for a biorthogonal signal set with

$M = 4$ is identical to that for four-phase PSK. *Hint*: A change in variables from U_1 and U_2 to $W_1 = U_1 + U_2$ and $W_2 = U_1 - U_2$ simplifies the problem.

4.16 The input $s(t)$ to a bandpass filter is

$$s(t) = \text{Re}\left[s_0(t)\, e^{j2\pi f_c t}\right]$$

where $s_0(t)$ is a rectangular pulse as shown in Figure P4.16(a).

1. Determine the output $y(t)$ of the bandpass filter for all $t \ge 0$ if the impulse response of the filter is
$$g(t) = \text{Re}\left[h(t)e^{j2\pi f_c t}\right]$$
where $h(t)$ is an exponential as shown in Figure P4.16(b).
2. Sketch the *equivalent lowpass output* of the filter.
3. When would you sample the output of the filter if you wished to have the maximum output at the sampling instant? What is the value of the maximum output?
4. Suppose that in addition to the input signal $s(t)$, there is additive white Gaussian noise
$$n(t) = \text{Re}\left[z(t)e^{j2\pi f_c t}\right]$$
where $R_z(\tau) = 2N_0\delta(\tau)$. At the sampling instant determined in part 3, the signal sample is corrupted by an additive Gaussian noise term. Determine its mean and variance.
5. What is the signal-to-noise ratio γ of the sampled output?
6. Determine the signal-to-noise ratio when $h(t)$ is the matched filter to $s(t)$, and compare this result with the value of γ obtained in part 5.

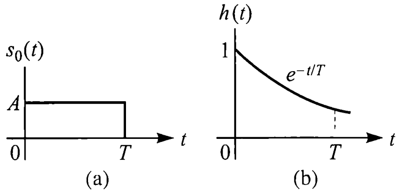


FIGURE P4.16

4.17 Consider the equivalent lowpass (complex-valued) signal $s_l(t)$, $0 \le t \le T$, with energy

$$\mathcal{E} = \int_0^T |s_l(t)|^2\, dt$$

Suppose that this signal is corrupted by AWGN, which is represented by its equivalent lowpass form $z(t)$. Hence, the observed signal is

$$r_l(t) = s_l(t) + z(t), \qquad 0 \le t \le T$$

The received signal is passed through a filter that has an (equivalent lowpass) impulse response $h_l(t)$. Determine $h_l(t)$ so that the filter maximizes the SNR at its output (at $t = T$).

4.18 In Section 3.2–4 it was shown that the minimum frequency separation for orthogonality of binary FSK signals with coherent detection is $\Delta f = 1/2T$. However, a lower error probability is possible with coherent detection of FSK if Δf is increased beyond $1/2T$. Show that the optimum value of Δf is $0.715/T$, and determine the probability of error for this value of Δf.

4.19 The equivalent lowpass waveforms for three signal sets are shown in Figure P4.19. Each set may be used to transmit one of four equally probable messages over an additive white

Gaussian noise channel. The equivalent lowpass noise $z(t)$ has zero-mean and autocorrelation function $R_z(\tau) = 2N_0\delta(\tau)$.

1. Classify the signal waveforms in sets I, II, III. In other words, state the category or class to which each signal set belongs.
2. What is the *average* transmitted energy for each signal set?
3. For signal set I, specify the average probability of error if the signals are detected coherently.
4. For signal set II, give a union bound on the probability of a symbol error if the detection is performed (i) coherently and (ii) noncoherently.
5. Is it possible to use noncoherent detection on signal set III? Explain.
6. Which signal set or signal sets would you select if you wished to achieve a spectral bit rate ($r = R/W$) of at least 2? Explain your answer.

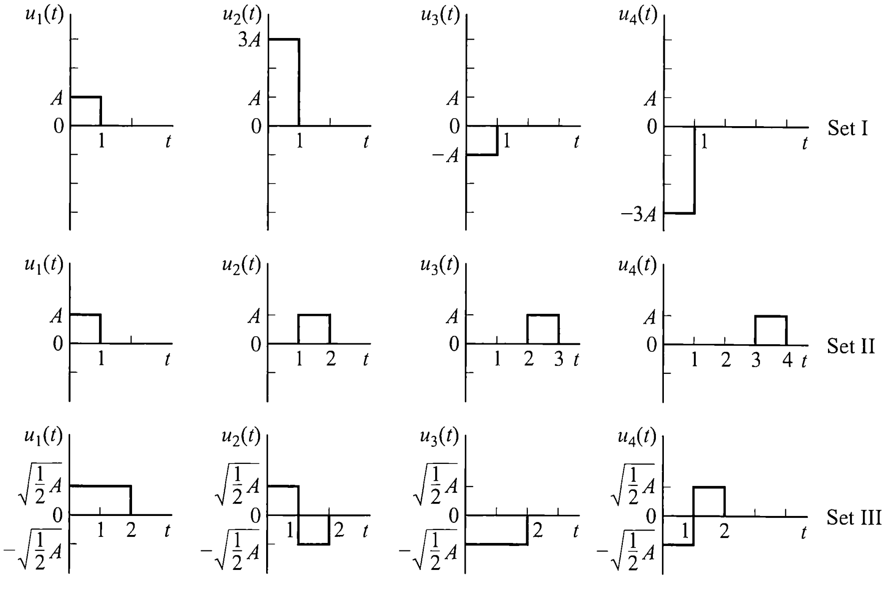

$u_1(t)$
A
0
1
t
$u_2(t)$
3A
A
0
1
t
$u_3(t)$
A
0
−A
1
t
$u_4(t)$
A
0
−3A
1
t
Set I
$u_1(t)$
A
0
1
t
$u_2(t)$
A
0
1 2 t
$u_3(t)$
A
0
1 2 3 t
$u_4(t)$
A
0
1 2 3 4 t
Set II
$u_1(t)$
$\sqrt{\frac{1}{2}}A$
0
$-\sqrt{\frac{1}{2}}A$
1 2 t
$u_2(t)$
$\sqrt{\frac{1}{2}}A$
0
$-\sqrt{\frac{1}{2}}A$
1 2 t
$u_3(t)$
$\sqrt{\frac{1}{2}}A$
0
$-\sqrt{\frac{1}{2}}A$
2 t
$u_4(t)$
$\sqrt{\frac{1}{2}}A$
0
$-\sqrt{\frac{1}{2}}A$
1 2 t
Set III

FIGURE P4.19

4.20 For the QAM signal constellation shown in Figure P4.20, determine the optimum decision boundaries for the detector, assuming that the SNR is sufficiently high that errors occur only between adjacent points.

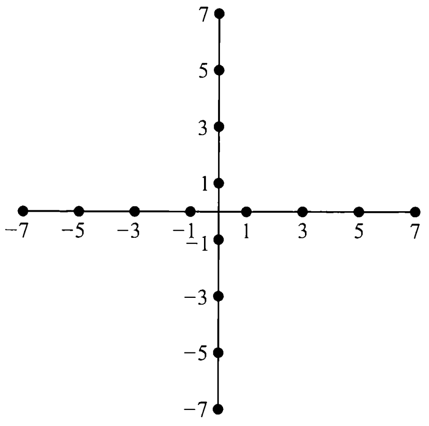
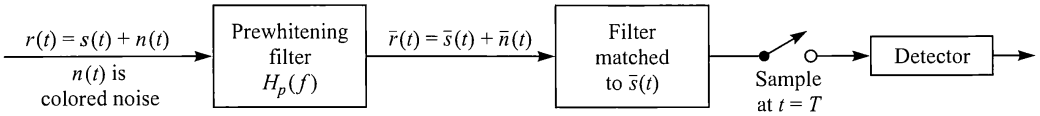

7
5
3
1
−7
−5
−3
−1
−1
1
3
5
7
−3
−5
−7

FIGURE P4.20

4.21 Two quadrature carriers $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$ are used to transmit digital information through an AWGN channel at two different data rates, 10 kilobits/s and 100 kilobits/s. Determine the relative amplitudes of the signals for the two carriers so that $\mathcal{E}_b/N_0$ for the two channels is identical.

4.22 When the additive noise at the input to the demodulator is colored, the filter matched to the signal no longer maximizes the output SNR. In such a case we may consider the use of a prefilter that "whitens" the colored noise. The prefilter is followed by a filter matched to the prefiltered signal. Toward this end, consider the configuration shown in Figure P4.22.

1. Determine the frequency response characteristic of the prefilter that whitens the noise, in terms of $s_n(f)$, the noise power spectral density.
2. Determine the frequency response characteristic of the filter matched to $\bar{s}(t)$.
3. Consider the prefilter and the matched filter as a single "generalized matched filter." What is the frequency response characteristic of this filter?
4. Determine the SNR at the input to the detector.



$r(t) = s(t) + n(t)$
$n(t)$ is colored noise
Prewhitening filter $H_p(f)$
$\bar{r}(t) = \bar{s}(t) + \bar{n}(t)$
Filter matched to $\bar{s}(t)$
Sample at $t = T$
Detector

FIGURE P4.22

4.23 Consider a digital communication system that transmits information via QAM over a voice-band telephone channel at a rate of 2400 symbols/s. The additive noise is assumed to be white and Gaussian.

1. Determine the $\mathcal{E}_b/N_0$ required to achieve an error probability of 10^{-5} at 4800 bits/s.
2. Repeat part 1 for a rate of 9600 bits/s.
3. Repeat part 1 for a rate of 19,200 bits/s.
4. What conclusions do you reach from these results?

4.24 Three equiprobable messages m_1, m_2, and m_3 are to be transmitted over an AWGN channel with noise power spectral density $\frac{1}{2}N_0$. The messages are

$$s_1(t) = \begin{cases} 1 & 0 \le t \le T \\ 0 & \text{otherwise} \end{cases} \qquad s_2(t) = -s_3(t) = \begin{cases} 1 & 0 \le t \le \frac{1}{2}T \\ -1 & \frac{1}{2}T < t \le T \\ 0 & \text{otherwise} \end{cases}$$

1. What is the dimensionality of the signal space?
2. Find an appropriate basis for the signal space.
3. Draw the signal constellation for this problem.
4. Derive and sketch the optimal decision regions R_1, R_2, and R_3.
5. Which of the three messages is most vulnerable to errors and why? In other words, which of $P(\text{error} \mid m_i \text{ transmitted})$, $i = 1, 2, 3$, is largest?

4.25 A QPSK communication system over an AWGN channel uses one of the four equiprobable signals $s_i(t) = A\cos(2\pi f_c t + i\pi/2)$, where $i = 0, 1, 2, 3$, f_c is the carrier frequency, and the duration of each signal is T. The power spectral density of the channel noise is $N_0/2$.

1. Express the message error probability of this system in terms of A, T, and N_0 (an approximate expression is sufficient).
2. If Gray coding is used, what is the bit error probability in terms of the same parameters used in part 1?
3. What is the minimum (theoretical minimum) required transmission bandwidth for this communication system?
4. If, instead of QPSK, binary FSK is used with $s_1(t) = B\cos 2\pi f_c t$ and $x_2(t) = B\cos(2\pi f_c + \Delta f)t$ where the duration of the signals is now T_1 and $\Delta f = \frac{1}{2T_1}$, determine the required T_1 and B in terms of T and A to achieve the same bit rate and the same bit error probability as the QPSK system described in parts 1–3.

4.26 A binary signaling scheme over an AWGN channel with noise power spectral density of $\frac{N_0}{2}$ uses the equiprobable messages shown in Figure P4.26 and is operating at a bit rate of R bits/s.

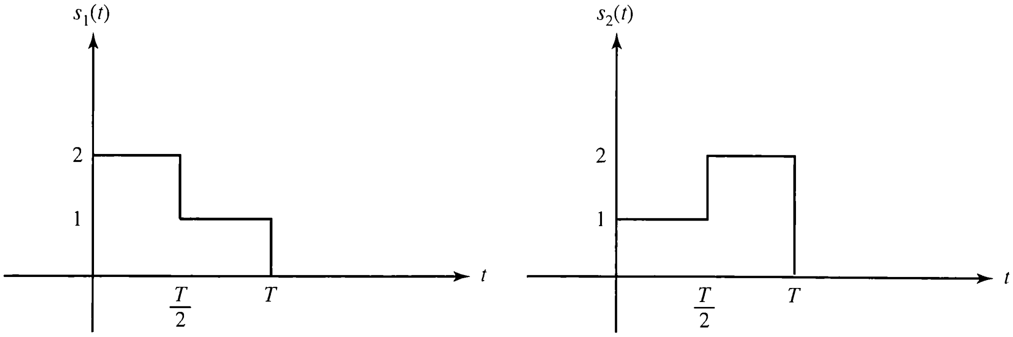


FIGURE P4.26

1. What is $\frac{\mathcal{E}_b}{N_0}$ for this system (in terms of N_0 and R)?
2. What is the error probability for this system (in terms of N_0 and R)?
3. By how many decibels does this system underperform a binary antipodal signaling system with the same $\frac{\mathcal{E}_b}{N_0}$?
4. Now assume that this system is augmented with two more signals $s_3(t) = -s_1(t)$ and $s_4(t) = -s_2(t)$ to result in a 4-ary equiprobable system. What is the resulting transmission bit rate?
5. Using the union bound, find a bound on the error probability of the 4-ary system introduced in part 4.

4.27 The four signals shown in Figure P4.27 are used for communication of four equiprobable messages over an AWGN channel. The noise power spectral density is $\frac{N_0}{2}$.

1. Find an orthonormal basis, with lowest possible N, for representation of the signals.
2. Plot the constellation, and *using the constellation,* find the energy in each signal. What is the average signal energy and what is $\mathcal{E}_{\text{bavg}}$?
3. On the constellation that you have plotted, determine the optimal decision regions for each signal, and determine which signal is more probable to be received in error.
4. Now analytically (i.e., *not* geometrically) determine the shape of the decision region for signal $s_1(t)$, i.e., D_1, and compare it with your result in part 3.

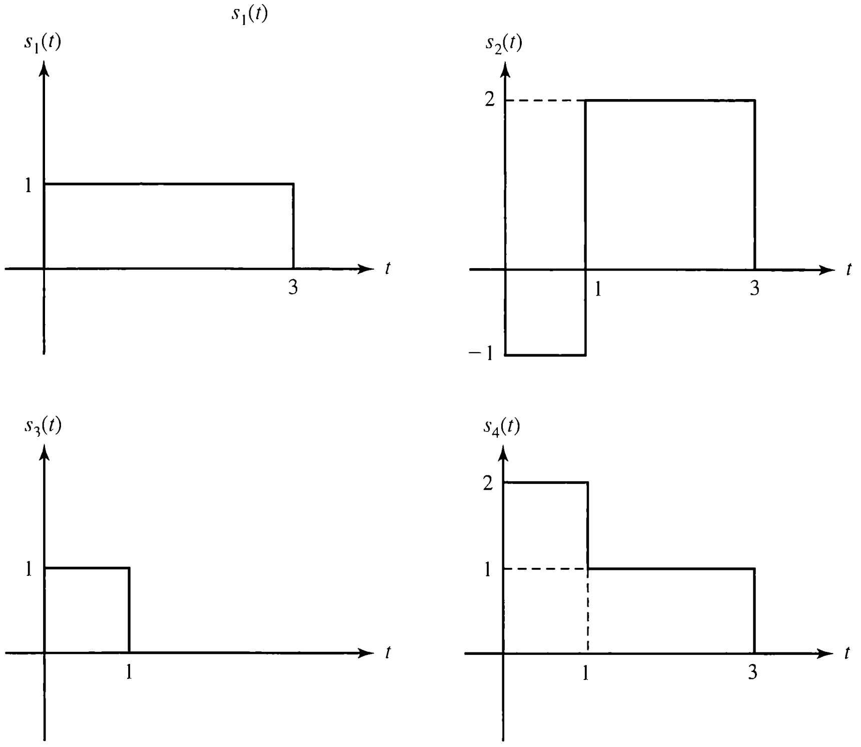


FIGURE P4.27

4.28 Consider the four-phase and eight-phase signal constellations shown in Figure P4.28. Determine the radii r_1 and r_2 of the circles such that the distance between two adjacent points in the two constellations is d. From this result, determine the additional transmitted energy required in the 8-PSK signal to achieve the same error probability as the four-phase signal at high SNR, where the probability of error is determined by errors in selecting adjacent points.

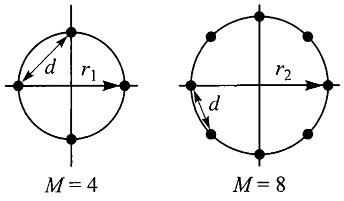


FIGURE P4.28

4.29 Digital information is to be transmitted by carrier modulation through an additive Gaussian noise channel with a bandwidth of 100 kHz and $N_0 = 10^{-10}$ W/Hz. Determine the maximum rate that can be transmitted through the channel for four-phase PSK, binary FSK, and four-frequency orthogonal FSK, which is detected noncoherently.

4.30 A continuous-phase FSK signal with $h = \frac{1}{2}$ is represented as

$$s(t) = \pm\sqrt{\frac{2\mathcal{E}_b}{T_b}}\cos\left(\frac{\pi t}{2T_b}\right)\cos 2\pi f_c t \pm \sqrt{\frac{2\mathcal{E}_b}{T_b}}\sin\left(\frac{\pi t}{2T_b}\right)\sin 2\pi f_c t, \qquad 0 \le t \le 2T_b$$

where the ± signs depend on the information bits transmitted.

1. Show that this signal has constant envelope.
2. Sketch a block diagram of the modulator for synthesizing the signal.
3. Sketch a block diagram of the demodulator and detector for recovering the information.

4.31 Consider a biorthogonal signal set with $M = 8$ signal points. Determine a union bound for the probability of a symbol error as a function of $\mathcal{E}_b/N_0$. The signal points are equally likely a priori.

4.32 Consider an M-ary digital communication system where $M = 2^N$, and N is the dimension of the signal space. Suppose that the M signal vectors lie on the vertices of a hypercube that is centered at the origin. Determine the average probability of a symbol error as a function of $\mathcal{E}_s/N_0$ where $\mathcal{E}_s$ is the energy per symbol, $\frac{1}{2}N_0$ is the power spectral density of the AWGN, and all signal points are equally probable.

4.33 Consider the signal waveform

$$s(t) = \sum_{i=1}^{n} c_i p(t - iT_c)$$

where $p(t)$ is a rectangular pulse of unit amplitude and duration T_c. The $\{c_i\}$ may be viewed as a code vector $\boldsymbol{c} = (c_1\ c_2\ \cdots\ c_n)$, where the elements $c_i = \pm 1$. Show that the filter matched to the waveform $s(t)$ may be realized as a cascade of a filter matched to $p(t)$ followed by a discrete-time filter matched to the vector $\mathbf{c}$. Determine the value of the output of the matched filter at the sampling instant $t = nT_c$.

4.34 A Hadamard matrix is defined as a matrix whose elements are ± 1 and whose row vectors are pairwise orthogonal. In the case when n is a power of 2, an $n \times n$ Hadamard matrix is constructed by means of the recursion given by Equation 3.2–59.

1. Let $\boldsymbol{c}_i$ denote the ith row of an $n \times n$ Hadamard matrix. Show that the waveforms constructed as

$$s_i(t) = \sum_{k=1}^{n} c_{ik}\, p(t - kT_c), \qquad i = 1, 2, \ldots, n$$

are orthogonal, where $p(t)$ is an arbitrary pulse confined to the time interval $0 \le t \le T_c$.

2. Show that the matched filters (or cross-correlators) for the n waveforms $\{s_i(t)\}$ can be realized by a single filter (or correlator) matched to the pulse $p(t)$ followed by a set of n cross-correlators using the code words $\{\boldsymbol{c}_i\}$.

4.35 The discrete sequence

$$r_k = \sqrt{\mathcal{E}}c_k + n_k, \qquad k = 1, 2, \ldots, n$$

represents the output sequence of samples from a demodulator, where $c_k = \pm 1$ are elements of one of two possible code words, $\boldsymbol{c}_1 = [1\ 1\ \cdots\ 1]$ and $\boldsymbol{c}_2 = [1\ 1\ \cdots\ 1\ -1\ \cdots\ -1]$. The code word $\boldsymbol{c}_2$ has w elements that are $+1$ and $n - w$ elements that are -1, where w is some positive integer. The noise sequence $\{n_k\}$ is white Gaussian with variance σ^2.

1. What is the optimum maximum-likelihood detector for the two possible transmitted signals?
2. Determine the probability of error as a function of the parameters $(\sigma^2, \mathcal{E}, w)$.
3. What is the value of w that minimizes the error?

4.36 In on-off keying of a carrier modulated signal, the two possible signals are

$$s_0(t) = 0, \qquad s_1(t) = \sqrt{\frac{2\mathcal{E}_b}{T_b}} \cos 2\pi f_c t, \qquad 0 \le t \le T_b$$

The corresponding received signals are

$$r(t) = n(t), \qquad 0 \le t \le T_b$$

$$r(t) = \sqrt{\frac{2\mathcal{E}_b}{T_b}} \cos(2\pi f_c t + \phi) + n(t), \qquad 0 \le t \le T_b$$

where ϕ is the carrier phase and $n(t)$ is AWGN.

1. Sketch a block diagram of the receiver (demodulator and detector) that employs noncoherent (envelope) detection.
2. Determine the PDFs for the two possible decision variables at the detector corresponding to the two possible received signals.
3. Derive the probability of error for the detector.

4.37 This problem deals with the characteristics of a DPSK signal.

1. Suppose we wish to transmit the data sequence

$$1\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 0$$

by binary DPSK. Let $s(t) = A\cos(2\pi f_c t + \theta)$ represent the transmitted signal in any signaling interval of duration T. Give the phase of the transmitted signal for the data sequence. Begin with $\theta = 0$ for the phase of the first bit to be transmitted.
2. If the data sequence is uncorrelated, determine and sketch the power density spectrum of the signal transmitted by DPSK.

4.38 In two-phase DPSK, the received signal in one signaling interval is used as a phase reference for the received signal in the following signaling interval. The decision variable is

$$D = \mathrm{Re}(V_m V_{m-1}^*) \underset{\text{“0”}}{\overset{\text{“1”}}{\gtrless}} 0$$

where

$$V_k = 2\mathcal{E} e^{(j\theta_k - \phi)} + N_k$$

represents the complex-valued output of the filter matched to the transmitted signal $u(t)$; N_k is a complex-valued Gaussian variable having zero mean and statistically independent components.

1. Writing $V_k = X_k + jY_k$, show that D is equivalent to

$$D = \left[\frac{1}{2}(X_m + X_{m-1})\right]^2 + \left[\frac{1}{2}(Y_m + Y_{m-1})\right]^2 - \left[\frac{1}{2}(X_m - X_{m-1})\right]^2 - \left[\frac{1}{2}(Y_m - Y_{m-1})\right]^2$$

2. For mathematical convenience; suppose that $\theta_k = \theta_{k-1}$. Show that the random variables U_1, U_2, U_3, and U_4 are statistically independent Gaussian variables, where $U_1 = \frac{1}{2}(X_m + X_{m-1})$, $U_2 = \frac{1}{2}(Y_m + Y_{m-1})$, $U_3 = \frac{1}{2}(X_m - X_{m-1})$, and $U_4 = \frac{1}{2}(Y_m - Y_{m-1})$.
3. Define the random variables $W_1 = U_1^2 + U_2^2$ and $W_2 = U_3^2 + U_4^2$. Then

$$D = W_1 - W_2 \underset{\text{“0”}}{\overset{\text{“1”}}{\gtrless}} 0$$

Determine the probability density functions for W_1 and W_2.

4. Determine the probability of error P_b, where

$$P_b = P(D < 0) = P(W_1 - W_2 < 0) = \int_0^{\infty} P(W_2 > w_1 | w_1) p(w_1)\, dw_1$$

4.39 Assuming that it is desired to transmit information at the rate of R bits/s, determine the required transmission bandwidth of each of the following six communication systems, and arrange them in order of bandwidth efficiency, starting from the most bandwidth-efficient and ending at the least bandwidth-efficient.

1. Orthogonal BFSK
2. 8PSK
3. QPSK
4. 64-QAM
5. BPSK
6. Orthogonal 16-FSK

4.40 In a binary communication system over an additive white Gaussian noise channel, two messages represented by antipodal signals $s_1(t)$ and $s_2(t) = -s_1(t)$ are transmitted. The probabilities of the two messages are p and $1 - p$, respectively, where $0 \leq p \leq 1/2$. The energy content of the each message is denoted by $\mathcal{E}$, and the noise power spectral density is $\frac{N_0}{2}$.

1. What is the expression for the threshold value r_{th} such that for $r > r_{\text{th}}$ the optimal detector makes a decision in favor of $s_1(t)$? What is the expression for the error probability?
2. Now assume that with probability of $1/2$ the link between the transmitter and the receiver is out of service and with a probability of $1/2$ this link remains in service. When the link is out of service, the receiver receives only noise. The receiver does not know whether the link is in service. What is the structure of the optimal receiver in this case? In particular, what is the value of the threshold r_{th} in this case? What is the value of the threshold if $p = 1/2$? What is the resulting error probability for this case $(p = 1/2)$?

4.41 A digital communication system with two equiprobable messages uses the following signals:

$$s_1(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 2 & 1 \leq t < 2 \\ 0 & \text{otherwise} \end{cases} \qquad s_2(t) = \begin{cases} 1 & 0 \leq t < 1 \\ -2 & 1 \leq t < 2 \\ 0 & \text{otherwise} \end{cases}$$

1. Assuming that the channel is AWGN with noise power spectral density $N_0/2$, determine the error probability of the optimal receiver and express it in terms of $\mathcal{E}_b/N_0$. By how many decibels does this system underperform a binary antipodal signaling system?
2. Assume that we are using the two-path channel shown in Figure P4.41

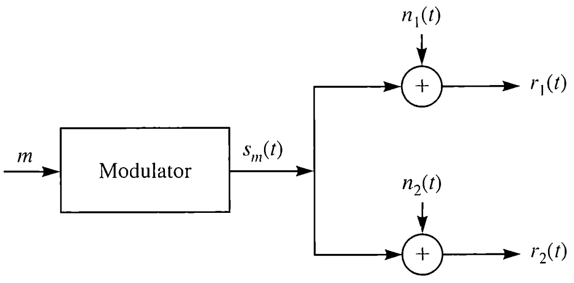


FIGURE P4.41

in which we receive both $r_1(t)$ and $r_2(t)$ at the receiver. Both $n_1(t)$ and $n_2(t)$ are *independent* white Gaussian processes each with power spectral density $N_0/2$. The receiver observes both $r_1(t)$ and $r_2(t)$ and makes its decision based on this observation. Determine the structure of the optimal receiver and the error probability in this case.

3. Now assume that $r_1(t) = As_m(t) + n_1(t)$ and $r_2(t) = s_m(t) + n_2(t)$, where m is the transmitted message and A is a random variable uniformly distributed over the interval $[0, 1]$. Assuming that the receiver *knows* the value of A, what is *his* optimal decision rule? What is the error probability in this case? (*Note*: This last question, regarding the error probability, is asked from *you*, and you *do not know* the value of A.)
4. If the receiver *does not know* the value of A, what is his optimal decision rule?

4.42 Two equiprobable messages m_1 and m_2 are to be transmitted through a channel with input X and output Y related by $Y = \rho X + N$, where N is a zero-mean Gaussian noise with variance σ^2 and ρ is a random variable independent of the noise.

1. Assuming an antipodal signaling scheme ($X = \pm A$) and a constant $\rho = 1$, what is the optimal decision rule and the resulting error probability?
2. With antipodal signaling, if ρ takes ± 1 with equal probability, what will be the optimal decision rule and the resulting error probability?
3. With antipodal signaling, if ρ takes 0 and 1 with equal probability, what will be the optimal decision rule and the resulting error probability?
4. Assuming an on-off signaling ($X = 0$ or A) and ρ taking ± 1 with equal probability, what will be the optimal decision rule?

4.43 A binary communication scheme uses two equiprobable messages $m = 1, 2$ corresponding to signals $s_1(t)$ and $s_2(t)$, where

$$\begin{aligned} s_1(t) &= x(t) \\ s_2(t) &= x(t-1) \end{aligned}$$

and $x(t)$ is shown Figure P4.43.

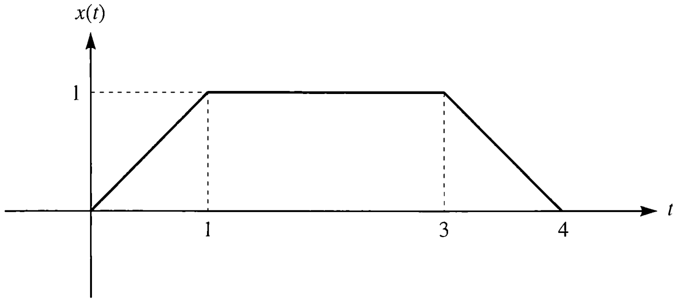


FIGURE P4.43

The power spectral density of the noise is $N_0/2$.

1. Design an optimal matched filter receiver for this system. Carefully label the diagram and determine all the required parameters.
2. Determine the error probability for this communication system.
3. Show that the receiver can be implemented using only *one* matched filter.
4. Now assume that $s_1(t) = x(t)$ and

$$s_2(t) = \begin{cases} x(t-1) & \text{with probability } 0.5 \\ x(t) & \text{with probability } 0.5 \end{cases}$$

In other words, in this case for $m = 1$ the transmitter always sends $x(t)$, but for $m = 2$ it is equally likely to send either $x(t)$ or $x(t-1)$. Determine the optimal detection rule for this case, and find the corresponding error probability.

4.44 Let X denote a Rayleigh distributed random variable, i.e.,

$$f_X(x) = \begin{cases} \dfrac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

1. Determine $\mathrm{E}\left[Q(\beta X)\right]$, where β is a positive constant. (*Hint*: Use the definition of the Q function and change the order of integration.)
2. In a binary antipodal signaling, let the received energy be subject to a Rayleigh distributed attenuation; i.e., let the received signal be $r(t) = \alpha s_m(t) + n(t)$, and therefore, $P_b = Q\left(\sqrt{\frac{2\alpha^2 \mathcal{E}_b}{N_0}}\right)$, where α^2 denotes the power attenuation and α has a Rayleigh PDF similar to X. Determine the average error probability of this system.
3. Repeat part 2 for a binary orthogonal system in which $P_b = Q\left(\sqrt{\frac{\alpha^2 \mathcal{E}_b}{N_0}}\right)$.
4. Find approximations for the results of parts 2 and 3 with the assumption that $\sigma^2 \frac{\mathcal{E}_b}{N_0} \gg 1$, and show that in this case both average error probabilities are proportional to $\frac{1}{\overline{\mathrm{SNR}}}$ where $\overline{\mathrm{SNR}} = 2\sigma^2 \frac{\mathcal{E}_b}{N_0}$.
5. Now find the average of $e^{-\beta\alpha^2}$, where β is a positive constant and α is a random variable distributed as $f_X(x)$. Find an approximation in this case when $\beta\sigma^2 \gg 1$. We will later see that this corresponds to the error probability of a noncoherent system in fading channels.

4.45 In a binary communication system two equiprobable messages $s_1 = (1, 1)$ and $s_2 = (-1, -1)$ are used. The received signal is $\boldsymbol{r} = \boldsymbol{s} + \boldsymbol{n}$, where $\boldsymbol{n} = (n_1, n_2)$. It is assumed that n_1 and n_2 are independent and each is distributed according to

$$f(n) = \frac{1}{2} e^{-|n|}$$

Determine and plot the decision regions D_1 and D_2 in this communication scheme.

4.46 Two equiprobable messages are transmitted via an additive white Gaussian noise channel with noise power spectral density of $\frac{N_0}{2} = 1$. The messages are transmitted by the following two signals

$$s_1(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

and $s_2(t) = s_1(t-1)$. It is intended to implement the receiver by using a correlation-type structure, but due to imperfections in the design of the correlators, the structure shown in Figure P4.46 has been implemented. The imperfection appears in the integrator in the upper branch where instead of $\int_0^1$ we have $\int_0^{1.5}$. The decision device, therefore, observes r_1 and r_2 and based on this observation has to decide which message was transmitted. What decision rule should be adopted by the decision device for an optimal decision?

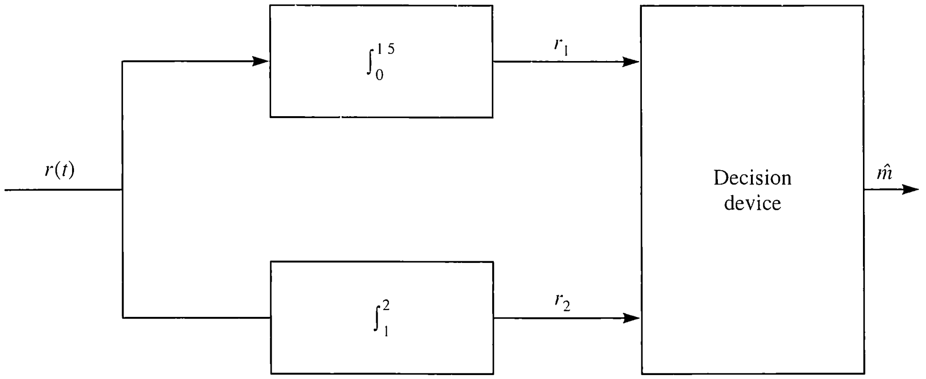
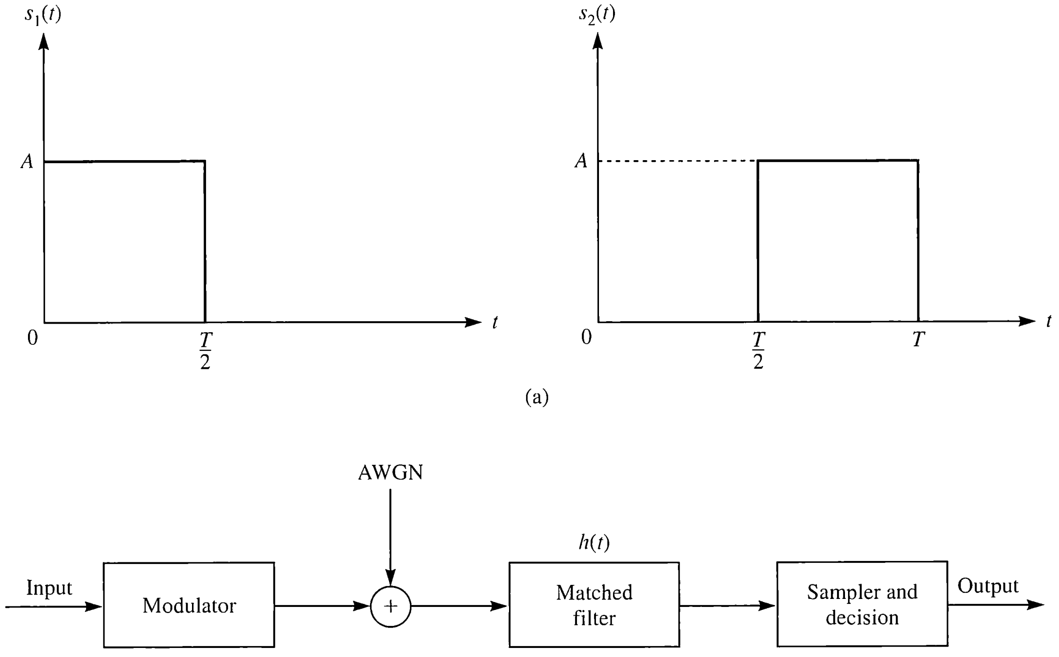

$\int_0^{1.5}$
$\int_1^2$
$r(t)$
r_1
r_2
Decision device
$\hat{m}$

FIGURE P4.46

4.47 A baseband digital communication system employs the signals shown in Figure P4.47(a) for transmission of two equiprobable messages. It is assumed the communication problem studied here is a "one-shot" communication problem; i.e., the above messages are transmitted just once, and no transmission takes place afterward. The channel has no attenuation, and the noise is AWG with power spectral density $\frac{N_0}{2}$.

1. Find an appropriate orthonormal basis for the representation of the signals.
2. In a block diagram, give the precise specifications of the optimal receiver using matched filters. Label the block diagram carefully.
3. Find the error probability of the optimal receiver.
4. Show that the optimal receiver can be implemented by using just *one* filter (see block diagram shown in Figure P4.47(b)). What are the characteristics of the matched filter and the sampler and decision device?


$s_1(t)$
A
0
$\frac{T}{2}$
t
$s_2(t)$
A
0
$\frac{T}{2}$
T
t
AWGN
$h(t)$
Input
Modulator
Matched filter
Sampler and decision
Output

(a)

(b)

FIGURE P4.47

5. Now assume the channel is not ideal, but has an impulse response of $c(t) = \delta(t) + \frac{1}{2}\delta(t - \frac{T}{2})$. Using the same matched filter you used in part 4, derive the optimal decision rule.
6. Assuming that the channel impulse response is $c(t) = \delta(t) + a\delta(t - \frac{T}{2})$, where a is a random variable uniformly distributed on $[0, 1]$, and using the same matched filter, derive the optimal decision rule.

4.48 A binary communication system uses antipodal signals $s_1(t) = s(t)$ and $s_2(t) = -s(t)$ for transmission of two equiprobable messages m_1 and m_2. The block diagram of the communication system is given in Figure P4.48.

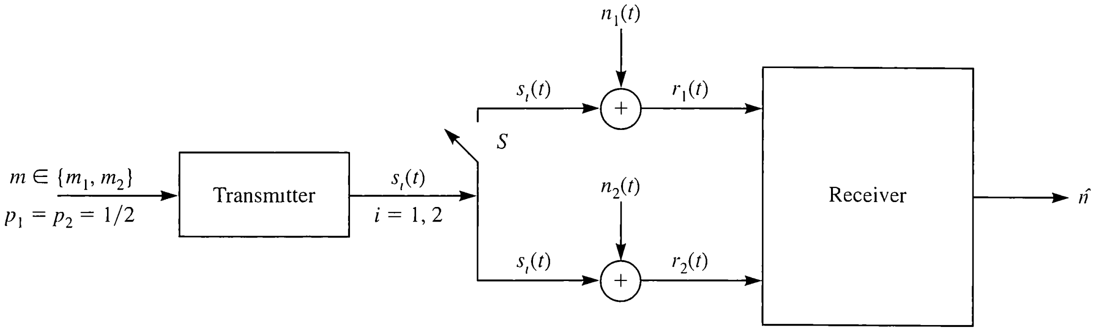


FIGURE P4.48

Message $s_i(t)$ is transmitted through *two* paths to a single receiver, and the receiver makes its decision based on the observation of *both* received signals $r_1(t)$ and $r_2(t)$. However, the upper channel is connected by a switch S which can either be closed or open. When the switch is open, $r_1(t) = n_1(t)$; i.e., the first channel provides only noise to the receiver. The switch is open or closed randomly with equal probability, but during the transmission it will not change position. Throughout this problem, it is assumed that the two noise processes are stationary, zero-mean, independent, white and Gaussian processes each with a power spectral density of $N_0/2$.

1. If the receiver does not know the position of the switch, determine the optimal decision rule.
2. Now assume that the receiver knows the position of the switch (the switch is still equally likely to be open or closed). What is the optimal decision rule in this case, and what is the resulting error probability?
3. In this part assume that *both the transmitter and the receiver* know the position of the switch (which is still equally likely to be open or closed). Assume that in this case the transmitter has a certain level of energy that it can transmit. To be more specific, assume that in the upper arm $\alpha s_i(t)$ and in the lower arm $\beta s_i(t)$ is transmitted, where $\alpha, \beta \geq 0$ and $\alpha^2 + \beta^2 = 2$. What is the best power allocation strategy by the transmitter (i.e., what is the best choice for α and β), what is the optimal decision rule at the receiver, and what is the resulting error probability?

4.49 The block diagram of a two-path communication system is shown in Figure P4.49. In the first path noise $n_1(t)$ is added to the transmitted signal. In the second path the signal is subject to a random amplification A and additive noise $n_2(t)$. The random variable A takes values ± 1 with equal probability. The transmitted signal is binary antipodal, and the two messages are equiprobable. Both $n_1(t)$ and $n_2(t)$ are zero-mean, white, Gaussian noise processes with power spectral densities $N_1/2$ and $N_2/2$, respectively. The receiver observes both $r_1(t)$ and $r_2(t)$.

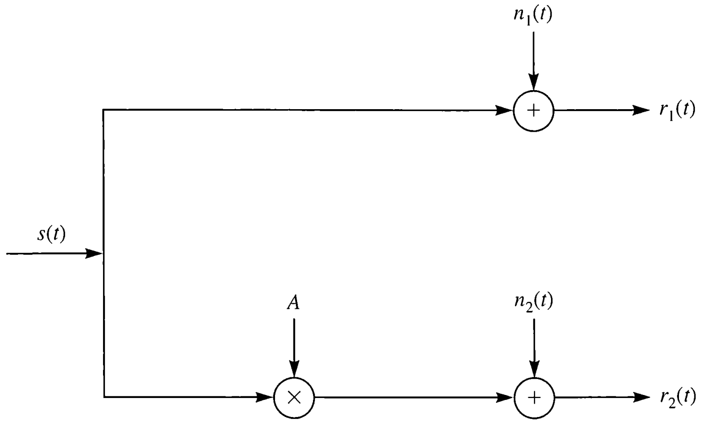


FIGURE P4.49

1. Assuming that the two noise processes are *independent*, determine the structure of the optimum receiver and find an expression for the error probability.
2. Now assume $N_1 = N_2 = 2$ and $E[n_1 n_2] = 1/2$, where n_1 and n_2 denote the projections of $n_1(t)$ and $n_2(t)$ on the unit signal in the direction of $s(t)$ (obviously the two noise processes are *dependent*). Determine the structure of the optimum receiver in this case.
3. What is the structure of the optimal receiver if the noise processes are independent and the receiver has access to $r(t) = r_1(t) + r_2(t)$ instead of observing $r_1(t)$ and $r_2(t)$ separately?
4. Determine the optimal decision rule if the two noise processes are independent and A can take 0 and 1 with equal probability [receiver has access to both $r_1(t)$ and $r_2(t)$].
5. What is the optimal detection rule in part 4 if we assume that the upper link is similar to the lower link but with A substituted with random variable B where $B = 1 - A$ (the lower link remains unchanged)?

4.50 A fading channel can be represented by the vector channel model $\boldsymbol{r} = a s_m + \boldsymbol{n}$, where a is a random variable denoting the fading, whose density function is given by the Rayleigh distribution

$$p(a) = \begin{cases} 2ae^{-a^2} & a \geq 0 \\ 0 & a < 0 \end{cases}$$

1. Assuming that equiprobable signals, binary antipodal signaling, and coherent detection are employed, what is the structure of the optimal receiver?
2. Show that the bit error probability in this case can be written as

$$P_b = \frac{1}{2}\left(1 - \sqrt{\frac{\mathcal{E}_b/N_0}{1 + E_b/N_0}}\right)$$

and for large SNR values we have

$$P_b \approx \frac{1}{4\mathcal{E}_b/N_0}$$

3. Assuming an error probability of 10^{-5} is desirable, determine the required SNR per bit (in dB) if (i) the channel is nonfading and (ii) the channel is a fading channel. How much more power is required by the fading channel to achieve the same bit error probability?

4. Show that if binary orthogonal signaling and noncoherent detection are employed, we have

$$P_b = \frac{1}{2 + \mathcal{E}_b/N_0}$$

4.51 A multiple access channel (MAC) is a channel with two transmitters and one receiver. The two transmitters transmit two messages, and the receiver is interested in correct detection of both messages. A block diagram of such system in the AWGN case is shown in Figure P4.51.

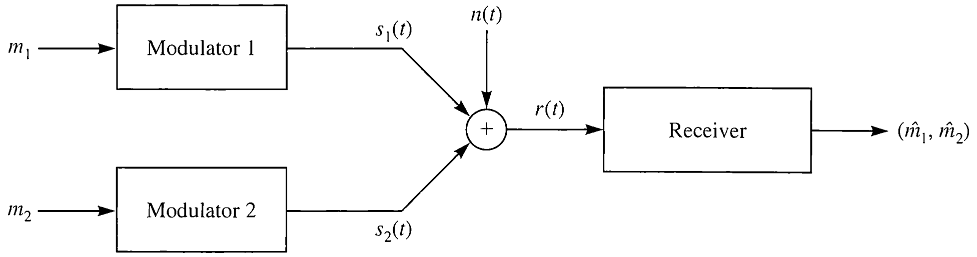


FIGURE P4.51

The messages are independent binary equiprobable random variables, and both modulators use binary antipodal signaling schemes. We have $s_1(t) = \pm g_1(t)$ and $s_2(t) = \pm g_2(t)$ depending on the values of m_1 and m_2, and $g_1(t)$ and $g_2(t)$ are two *unit energy* pulses each with duration T ($g_1(t)$ and $g_2(t)$ are not necessarily orthogonal). The received signal is $r(t) = s_1(t) + s_2(t) + n(t)$, where $n(t)$ is a white Gaussian process with a power spectral density of $N_0/2$.

1. What is the structure of the receiver that minimizes $P(\hat{m}_1 \neq m_1)$ *and* $P(\hat{m}_2 \neq m_2)$?
2. What is the structure of the receiver that minimizes $P((\hat{m}_1, \hat{m}_2) \neq (m_1, m_2))$?
3. Between receivers designed in parts 1 and 2, which would you label as the real optimal receiver? Which has a simpler structure?
4. What are the minimum error probabilities p_1 and p_2 for the receiver in part 1 and p_{12} for the receiver in part 2?

4.52 The constellation for an MPSK modulation system is shown in Figure P4.52. Only point s_1 and its decision region are shown here. The shaded area (extended to infinity) shows the error region when s_1 is transmitted.

1. Express R in terms of $\mathcal{E}$, θ, and M.
2. Using the value of R and integrating over the gray area, show that the error probability for this system can be written as

$$P_e = \frac{1}{\pi} \int_0^{\pi - \frac{\pi}{M}} e^{-\frac{\mathcal{E}}{N_0} \frac{\sin^2 \frac{\pi}{M}}{\sin^2 \theta}} \, d\theta$$

3. Find the error probability for $M = 2$, and by equating it with the error probability of BPSK, conclude that $Q(x)$ can be expressed as

$$Q(x) = \frac{1}{\pi} \int_0^{\frac{\pi}{2}} e^{-\frac{x^2}{2 \sin^2 \theta}} \, d\theta$$

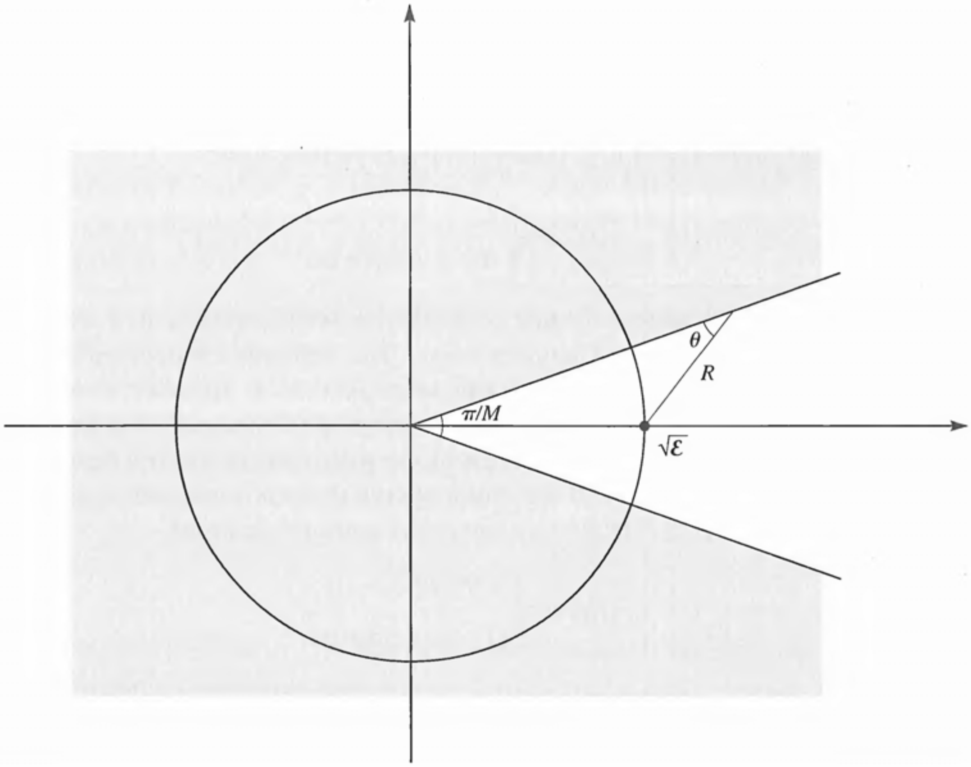


FIGURE P4.52

4.53 A communication system employs M signals $\{s_m(t)\}_{m=1}^{M}$ for transmission of M equiprobable messages. The receiver has two antennas and receives two signals $r_1(t) = s_m(t)+n_1(t)$ and $r_2(t) = s_m(t) + n_2(t)$ by these antennas. Both $n_1(t)$ and $n_2(t)$ are white Gaussian noises with power spectral densities $N_{01}/2$ and $N_{02}/2$, respectively. The receiver makes its optimal detection based on the observation of both $r_1(t)$ and $r_2(t)$. It is further assumed that the two noise processes are independent.

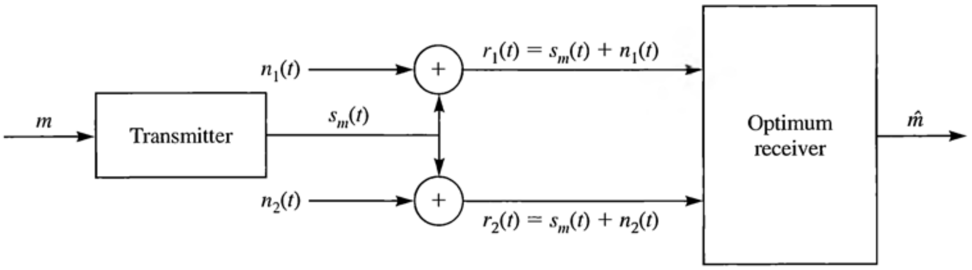


FIGURE P4.53

1. Determine the optimal decision rule for this receiver.
2. Assuming $N_{01} = N_{02} = N_0$, determine the optimal receiver structure.
3. Show that under the assumption of part 2, the receiver needs to know only $r_1(t)+r_2(t)$.
4. Now assume the system is binary and employs on-off signaling (i.e., $s_1(t) = s(t)$ and $s_2(t) = 0$), and show that the optimal decision rule consists of comparing $r_1 + \alpha r_2$ with a threshold. Determine α and the threshold (in this part you are assuming noise powers are different).

5. Show that in part 4, if noise powers are equal, then $\alpha = 1$, and determine the error probability in this case. How does this system compare with a system that has only one antenna, i.e., receives only $r_1(t)$?

4.54 A communication system employs binary antipodal signals with

$$s_1(t) = \begin{cases} 1 & 0 < t < 1 \\ 0 & \text{otherwise} \end{cases}$$

and $s_2(t) = -s_1(t)$. The received signal consists of a direct component, a scattered component, and the additive white Gaussian noise. The scattered component is a delayed version of the basic signal times a random amplification A. In other words, we have $r(t) = s(t) + As(t-1) + n(t)$, where $s(t)$ is the transmitted message, A is an exponential random variable, and $n(t)$ is a white Gaussian noise with a power spectral density of $N_0/2$. It is assumed that the time delay of the multipath component is constant (equal to 1) and A and $n(t)$ are independent. The two messages are equiprobable and

$$f_A(a) = \begin{cases} e^{-a} & a > 0 \\ 0 & \text{otherwise} \end{cases}$$

1. What is the optimal decision rule for this problem? Simplify the resulting rule as much as you can.
2. How does the error probability of this system compare with the error probability of a system which does not involve multipath? Which one has a better performance?

4.55 A binary communication system uses equiprobable signals $s_1(t)$ and $s_2(t)$

$$s_1(t) = \sqrt{2\mathcal{E}_b}\,\phi_1(t)\cos(2\pi f_c t)$$
$$s_2(t) = \sqrt{2\mathcal{E}_b}\,\phi_2(t)\cos(2\pi f_c t)$$

for transmission of two equiprobable messages. It is assumed that $\phi_1(t)$ and $\phi_2(t)$ are orthonormal. The channel is AWGN with noise power spectral density of $N_0/2$.

1. Determine the optimal error probability for this system, using a coherent detector.
2. Assuming that the demodulator has a phase ambiguity between 0 and θ ($0 \le \theta \le \pi$) in carrier recovery, and employs the same detector as in part 1, what is the resulting worst-case error probability?
3. What is the answer to part 2 in the special case where $\theta = \pi/2$?

4.56 In this problem we show that the volume of an n-dimensional sphere with radius R, defined by the set of all $\boldsymbol{x} \in \mathbb{R}^n$ such that $\|\boldsymbol{x}\| \le R$, is given by $V_n(R) = B_n R^n$, where

$$B_n = \frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}+1\right)}$$

1. Using change of variables, show that

$$V_n(R) = \underset{x_1^2+x_2^2+\ \ +x_n^2 \le R^2}{\int\int\ldots\int} dx_1\,dx_2\ldots dx_n = B_n R^n$$

where B_n is the volume on an n-dimensional sphere of radius 1, i.e., $B_n = V(1)$.

2. Consider n iid Gaussian random variables $Y_i, i = 1, 2, \ldots, n$, each distributed according to $\mathcal{N}(0, 1)$. Show that the probability that $\boldsymbol{Y} = (Y_1, Y_2, \ldots, Y_n)$ would lie in the area between two spheres of radii R and $R - \epsilon$, where $\epsilon > 0$ is very small such that $\frac{\epsilon}{R} \ll \frac{1}{n}$, can be approximated as

$$\begin{aligned} \mathrm{P}[R-\epsilon \leq \|\boldsymbol{Y}\| \leq R] &= p(\boldsymbol{y})\,[V_n(R) - V_n(R-\epsilon)] \\ &\approx \frac{\epsilon n R^{n-1} B_n}{(2\pi)^{n/2}}\, e^{-\frac{R^2}{2}} \end{aligned}$$

3. Note that $p(\boldsymbol{y})$ is a function of $\|\boldsymbol{y}\|$. From this show that we can also approximate $\mathrm{P}[R-\epsilon \leq |\boldsymbol{Y}\| \leq R]$ as

$$\mathrm{P}[R-\epsilon \leq \|\boldsymbol{Y}\| \leq R] \approx p_{\|\boldsymbol{Y}\|}(R)\epsilon$$

where $p_{\|\boldsymbol{Y}\|}(\cdot)$ denoted the PDF of $\|\boldsymbol{Y}\|$.

4. From parts 2 and 3 conclude that

$$p_{\|\boldsymbol{Y}\|}(r) = \frac{n r^{n-1} B_n}{(2\pi)^{n/2}}\, e^{-\frac{r^2}{2}}$$

5. Using the fact that $p_{\|\boldsymbol{Y}\|}(r)$ is a PDF and therefore its integral over the positive real line is equal to 1, conclude that

$$\frac{n B_n}{(2\pi)^{n/2}} \int_0^\infty r^{n-1}\, e^{-\frac{r^2}{2}}\, dr = 1$$

6. Using the definition of the gamma function given by Equation 2.3–22 as

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t}\, dt, \qquad x > 0$$

show that

$$\int_0^\infty r^{n-1}\, e^{-\frac{r^2}{2}}\, dr = 2^{\left(\frac{n}{2}-1\right)}\Gamma\left(\frac{n}{2}\right)$$

and conclude that

$$B_n = \frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}+1\right)}$$

4.57 Let $\mathbb{Z}^n + \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right)$ denote the n-dimensional integer lattice shifted by $1/2$, and let $\mathcal{R}$ be an n-dimensional hypercube centered at the origin with side length L which defines the boundary of this lattice. We further assume that n is even and $L = 2^\ell$ is a power of 2; the number of bits per two dimensions is denoted by β, and we consider a constellation $\mathcal{C}$ based on the intersection of the shifted lattice $\mathbb{Z}^n + \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right)$ and the boundary region $\mathcal{R}$ defined as an n-dimensional hypercube centered at the origin with side length L.

1. Show that $\beta = 2\ell + 2$.
2. Show that for this constellation the figure of merit is approximated by

$$\mathrm{CFM}(\mathcal{C}) \approx \frac{6}{2^\beta}$$

Note that this is equal to the CFM for a square QAM constellation.

3. Show the shaping gain of $\mathcal{R}$ is given by $\gamma_s(\mathcal{R}) = 1$.

4.58 Recall that MSK can be represented as a four-phase offset PSK modulation having the lowpass equivalent form

$$v(t) = \sum_k [I_k u(t - 2kT_b) + jJ_k u(t - 2kT_b - T_b)]$$

where

$$u(t) = \begin{cases} \sin(\pi t/2T_b) & 0 \le t \le 2T_b \\ 0 & \text{otherwise} \end{cases}$$

and $\{I_k\}$ and $\{J_k\}$ are sequences of information symbols (± 1).

1. Sketch the block diagram of an MSK demodulator for offset QPSK.
2. Evaluate the performance of the four-phase demodulator for AWGN if no account is taken of the memory in the modulation.
3. Compare the performance obtained in part 2 with that for Viterbi decoding of the MSK signal.
4. The MSK signal is also equivalent to binary FSK. Determine the performance of noncoherent detection of the MSK signal. Compare your result with parts 2 and 3.

4.59 Consider a transmission line channel that employs $n - 1$ regenerative repeaters plus the terminal receiver in the transmission of binary information. Assume that the probability of error at the detector of each receiver is p and that errors among repeaters are statistically independent.

1. Show that the binary error probability at the terminal receiver is

$$P_n = \frac{1}{2}\left[1 - (1 - 2p)^n\right]$$

2. If $p = 10^{-6}$ and $n = 100$, determine an approximate value of P_n.

4.60 A digital communication system consists of a transmission line with 100 digital (regenerative) repeaters. Binary antipodal signals are used for transmitting the information. If the overall end-to-end error probability is 10^{-6}, determine the probability of error for each repeater and the required $\mathcal{E}_b/N_0$ to achieve this performance in AWGN.

4.61 A radio transmitter has a power output of $P_T = 1$ W at a frequency of 1 GHz. The transmitting and receiving antennas are parabolic dishes with diameter $D = 3$ m.

1. Determine the antenna gains.
2. Determine the EIRP for the transmitter.
3. The distance (free space) between the transmitting and receiving antennas is 20 km. Determine the signal power at the output of the receiving antenna in decibels.

4.62 A radio communication system transmits at a power level of 0.1 W at 1 GHz. The transmitting and receiving antennas are parabolic, each having a diameter of 1 m. The receiver is located 30 km from the transmitter.

1. Determine the gains of the transmitting and receiving antennas.
2. Determine the EIRP of the transmitted signal.
3. Determine the signal power from the receiving antenna.

4.63 A satellite in synchronous orbit is used to communicate with an earth station at a distance of 40,000 km. The satellite has an antenna with a gain of 15 dB and a transmitter power of 3 W. The earth station uses a 10-m parabolic antenna with an efficiency of 0.6. The frequency band is at $f = 1$ GHz. Determine the received power level at the output of the receiver antenna.

4.64 A spacecraft located 100,000 km from the earth is sending data at a rate of R bits/s. The frequency band is centered at 2 GHz, and the transmitted power is 10 W. The earth station uses a parabolic antenna, 50 m in diameter, and the spacecraft has an antenna with a gain of 10 dB. The noise temperature of the receiver front end is $T_0 = 300$ K.

1. Determine the received power level.
2. If the desired $\mathcal{E}_b/N_0 = 10$ dB, determine the maximum bit rate that the spacecraft can transmit.

4.65 A satellite in geosynchronous orbit is used as a regenerative repeater in a digital communication system. Consider the satellite-to-earth link in which the satellite antenna has a gain of 6 dB and the earth station antenna has a gain of 50 dB. The downlink is operated at a center frequency of 4 GHz, and the signal bandwidth is 1 MHz. If the required $\mathcal{E}_b/N_0$ for reliable communication is 15 dB, determine the transmitted power for the satellite downlink. Assume that $N_0 = 4.1 \times 10^{-21}$ W/Hz.

5

Carrier and Symbol Synchronization

We have observed that in a digital communication system, the output of the demodulator must be sampled periodically, once per symbol interval, in order to recover the transmitted information. Since the propagation delay from the transmitter to the receiver is generally unknown at the receiver, symbol timing must be derived from the received signal in order to synchronously sample the output of the demodulator.

The propagation delay in the transmitted signal also results in a carrier offset, which must be estimated at the receiver if the detector is phase-coherent. In this chapter, we consider methods for deriving carrier and symbol synchronization at the receiver.

5.1

SIGNAL PARAMETER ESTIMATION

Let us begin by developing a mathematical model for the signal at the input to the receiver. We assume that the channel delays the signals transmitted through it and corrupts them by the addition of Gaussian noise. Hence, the received signal may be expressed as

$$r(t) = s(t-\tau) + n(t)$$

where

$$s(t) = \mathrm{Re}\left[s_l(t)e^{j2\pi f_c t}\right] \tag{5.1–1}$$

and where τ is the propagation delay and $s_l(t)$ is the equivalent low-pass signal.

The received signal may be expressed as

$$r(t) = \mathrm{Re}\left\{\left[s_l(t-\tau)e^{j\phi} + z(t)\right]e^{j2\pi f_c t}\right\} \tag{5.1–2}$$

where the carrier phase ϕ, due to the propagation delay τ, is $\phi = -2\pi f_c \tau$. Now, from this formulation, it may appear that there is only one signal parameter to be estimated, namely, the propagation delay, since one can determine ϕ from knowledge of f_c and τ. However, this is not the case. First of all, the oscillator that generates the carrier signal for demodulation at the receiver is generally not synchronous in phase with that at the transmitter. Furthermore, the two oscillators may be drifting slowly with time, perhaps in different directions. Consequently, the received carrier phase is not only dependent on the time delay τ. Furthermore, the precision to which one must synchronize in time for the purpose of demodulating the received signal depends on the symbol interval T. Usually, the estimation error in estimating τ must be a relatively small fraction of T. For example, ± 1 percent of T is adequate for practical applications. However, this level of precision is generally inadequate for estimating the carrier phase, even if ϕ depends only on τ. This is due to the fact that f_c is generally large, and, hence, a small estimation error in τ causes a large phase error.

In effect, we must estimate both parameters τ and ϕ in order to demodulate and coherently detect the received signal. Hence, we may express the received signal as

$$r(t) = s(t; \phi, \tau) + n(t) \tag{5.1–3}$$

where ϕ and τ represent the signal parameters to be estimated. To simplify the notation, we let $\boldsymbol{\theta}$ denote the parameter vector $\{\phi, \tau\}$, so that $s(t; \phi, \tau)$ is simply denoted by $s(t; \boldsymbol{\theta})$.

There are basically two criteria that are widely applied to signal parameter estimation: the *maximum-likelihood* (ML) criterion and the *maximum a posteriori probability* (MAP) criterion. In the MAP criterion, the signal parameter vector $\boldsymbol{\theta}$ is modeled as random and characterized by an a priori probability density function $p(\boldsymbol{\theta})$. In the maximum-likelihood criterion, the signal parameter vector $\boldsymbol{\theta}$ is treated as deterministic but unknown.

By performing an orthonormal expansion of $r(t)$ using N orthonormal functions $\{\phi_n(t)\}$, we may represent $r(t)$ by the vector of coefficients $(r_1\, r_2 \cdots r_N) \equiv \boldsymbol{r}$. The joint PDF of the random variables $(r_1\, r_2 \cdots r_N)$ in the expansion can be expressed as $p(\boldsymbol{r}|\boldsymbol{\theta})$. Then, the ML estimate of $\boldsymbol{\theta}$ is the value that maximizes $p(\boldsymbol{r}|\boldsymbol{\theta})$. On the other hand, the MAP estimate is the value of $\boldsymbol{\theta}$ that maximizes the a posteriori probability density function

$$p(\boldsymbol{\theta}|\boldsymbol{r}) = \frac{p(\boldsymbol{r}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{r})} \tag{5.1–4}$$

We note that if there is no prior knowledge of the parameter vector $\boldsymbol{\theta}$, we may assume that $p(\boldsymbol{\theta})$ is uniform (constant) over the range of values of the parameters. In such a case, the value of $\boldsymbol{\theta}$ that maximizes $p(\boldsymbol{r}|\boldsymbol{\theta})$ also maximizes $p(\boldsymbol{\theta}|\boldsymbol{r})$. Therefore, the MAP and ML estimates are identical.

In our treatment of parameter estimation given below, we view the parameters ϕ and τ as unknown, but deterministic. Hence, we adopt the ML criterion for estimating them.

In the ML estimation of signal parameters, we require that the receiver extract the estimate by observing the received signal over a time interval $T_0 \geq T$, which is called the observation interval. Estimates obtained from a single observation interval are sometimes called one-shot estimates. In practice, however, the estimation is performed on a continuous basis by using tracking loops (either analog or digital) that continuously update the estimates. Nevertheless, one-shot estimates yield insight for tracking loop implementation. In addition, they prove useful in the analysis of the performance of ML estimation, and their performance can be related to that obtained with a tracking loop.

5.1–1 The Likelihood Function

Although it is possible to derive the parameter estimates based on the joint PDF of the random variables $(r_1\, r_2 \cdots r_N)$ obtained from the expansion of $r(t)$, it is convenient to deal directly with the signal waveforms when estimating their parameters. Hence, we shall develop a continuous-time equivalent of the maximization of $p(\boldsymbol{r}\,|\,\boldsymbol{\theta})$.

Since the additive noise $n(t)$ is white and zero-mean Gaussian, the joint PDF $p(\boldsymbol{r}\,|\,\boldsymbol{\theta})$ may be expressed as

$$p(\boldsymbol{r}\,|\,\boldsymbol{\theta}) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^N \exp\left\{-\sum_{n=1}^{N}\frac{[r_n - s_n(\boldsymbol{\theta})]^2}{2\sigma^2}\right\} \tag{5.1–5}$$

where

$$\begin{aligned} r_n &= \int_{T_0} r(t)\phi_n(t)\,dt \\ s_n(\boldsymbol{\theta}) &= \int_{T_0} s(t;\,\boldsymbol{\theta})\phi_n(t)\,dt \end{aligned} \tag{5.1–6}$$

where T_0 represents the integration interval in the expansion of $r(t)$ and $s(t;\,\boldsymbol{\theta})$.

We note that the argument in the exponent may be expressed in terms of the signal waveforms $r(t)$ and $s(t;\,\boldsymbol{\theta})$, by substituting from Equation 5.1–6 into Equation 5.1–5. That is,

$$\lim_{N\to\infty}\frac{1}{2\sigma^2}\sum_{n=1}^{N}[r_n - s_n(\boldsymbol{\theta})]^2 = \frac{1}{N_0}\int_{T_0}[r(t) - s(t;\,\boldsymbol{\theta})]^2\,dt \tag{5.1–7}$$

where the proof is left as an exercise for the reader (see Problem 5.1). Now, the maximization of $p(\boldsymbol{r}\,|\,\boldsymbol{\theta})$ with respect to the signal parameters $\boldsymbol{\theta}$ is equivalent to the maximization of the *likelihood function*.

$$\Lambda(\boldsymbol{\theta}) = \exp\left\{-\frac{1}{N_0}\int_{T_0}[r(t) - s(t;\,\boldsymbol{\theta})]^2\,dt\right\} \tag{5.1–8}$$

Below, we shall consider signal parameter estimation from the viewpoint of maximizing $\Lambda(\boldsymbol{\theta})$.

5.1–2 Carrier Recovery and Symbol Synchronization in Signal Demodulation

Symbol synchronization is required in every digital communication system which transmits information synchronously. Carrier recovery is required if the signal is detected coherently.

Figure 5.1–1 illustrates the block diagram of a binary PSK (or binary PAM) signal demodulator and detector. As shown, the carrier phase estimate $\hat{\phi}$ is used in generating the reference signal $g(t)\cos(2\pi f_c t + \hat{\phi})$ for the correlator. The symbol synchronizer

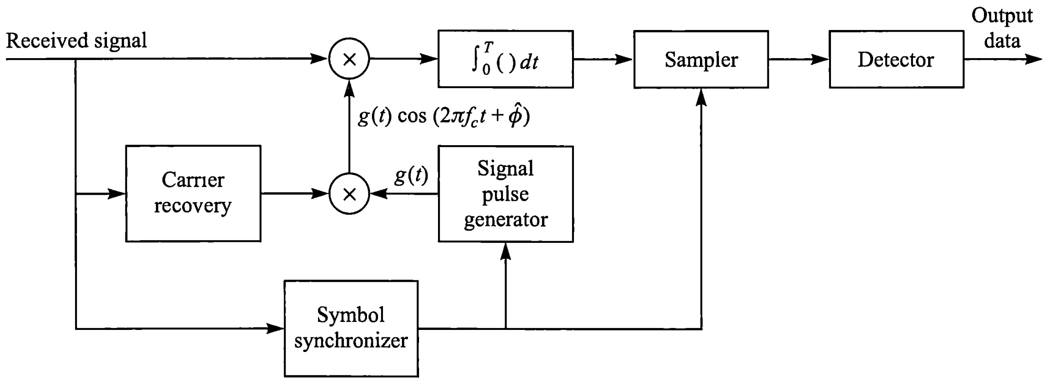

Received signal
$\int_0^T ()\,dt$
Sampler
Detector
Output data
$g(t)\cos(2\pi f_c t+\hat{\phi})$
Carrier recovery
$g(t)$
Signal pulse generator
Symbol synchronizer

FIGURE 5.1–1
Block diagram of a binary PSK receiver.

controls the sampler and the output of the signal pulse generator. If the signal pulse is rectangular, then the signal generator can be eliminated.

The block diagram of an M-ary PSK demodulator is shown in Figure 5.1–2. In this case, two correlators (or matched filters) are required to correlate the received signal with the two quadrature carrier signals $g(t)\cos(2\pi f_c t+\hat{\phi})$ and $g(t)\sin(2\pi f_c t+\hat{\phi})$, where $\hat{\phi}$ is the carrier phase estimate. The detector is now a phase detector, which compares the received signal phases with the possible transmitted signal phases.

The block diagram of a PAM signal demodulator is shown in Figure 5.1–3. In this case, a single correlator is required, and the detector is an amplitude detector, which

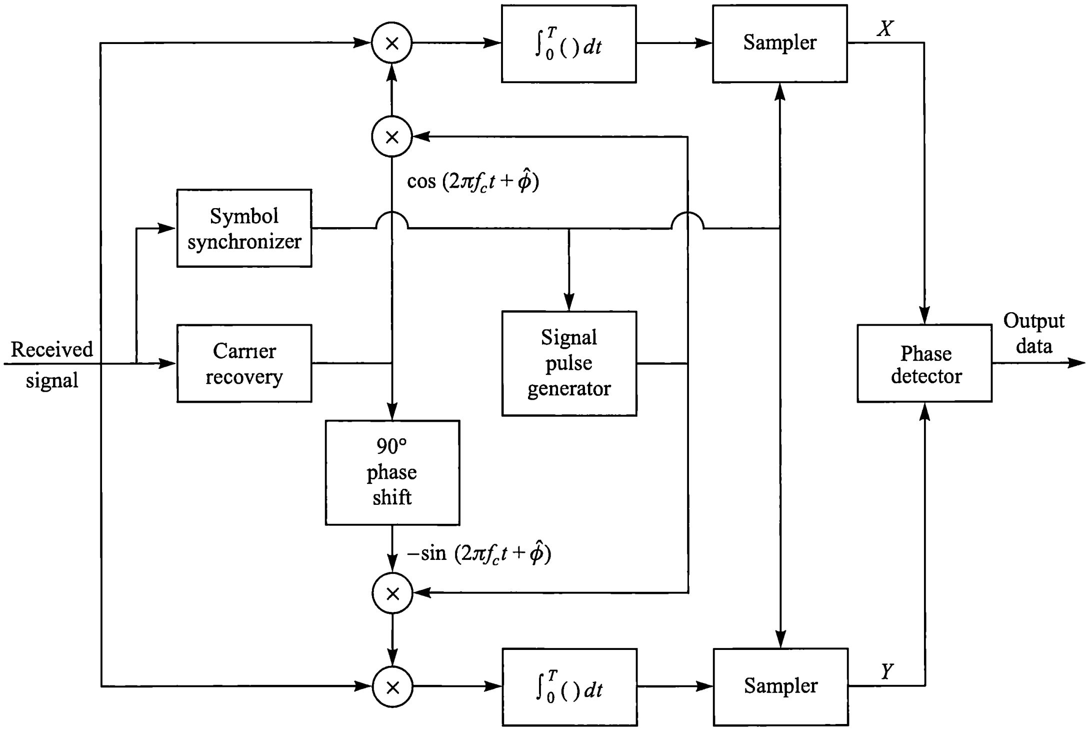

$\int_0^T ()\,dt$
Sampler
X
$\cos(2\pi f_c t+\hat{\phi})$
Symbol synchronizer
Received signal
Carrier recovery
Signal pulse generator
Phase detector
Output data
90° phase shift
$-\sin(2\pi f_c t+\hat{\phi})$
$\int_0^T ()\,dt$
Sampler
Y

FIGURE 5.1–2
Block diagram of an M-ary PSK receiver.

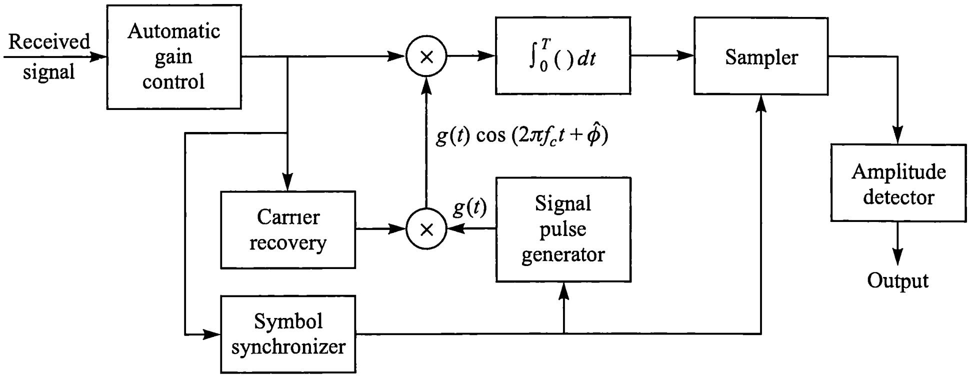

Received signal
Automatic gain control
$\int_0^T (\,)dt$
Sampler
$g(t)\cos(2\pi f_c t+\hat{\phi})$
$g(t)$
Carrier recovery
Signal pulse generator
Amplitude detector
Output
Symbol synchronizer

FIGURE 5.1–3
Block diagram of an M-ary PAM receiver.

compares the received signal amplitude with the possible transmitted signal amplitudes. Note that we have included an automatic gain control (AGC) at the front end of the demodulator to eliminate channel gain variations, which would affect the amplitude detector. The AGC has a relatively long time constant, so that it does not respond to the signal amplitude variations that occur on a symbol-by-symbol basis. Instead, the AGC maintains a fixed average (signal plus noise) power at its output.

Finally, we illustrate the block diagram of a QAM demodulator in Figure 5.1–4. As in the case of PAM, an AGC is required to maintain a constant average power signal

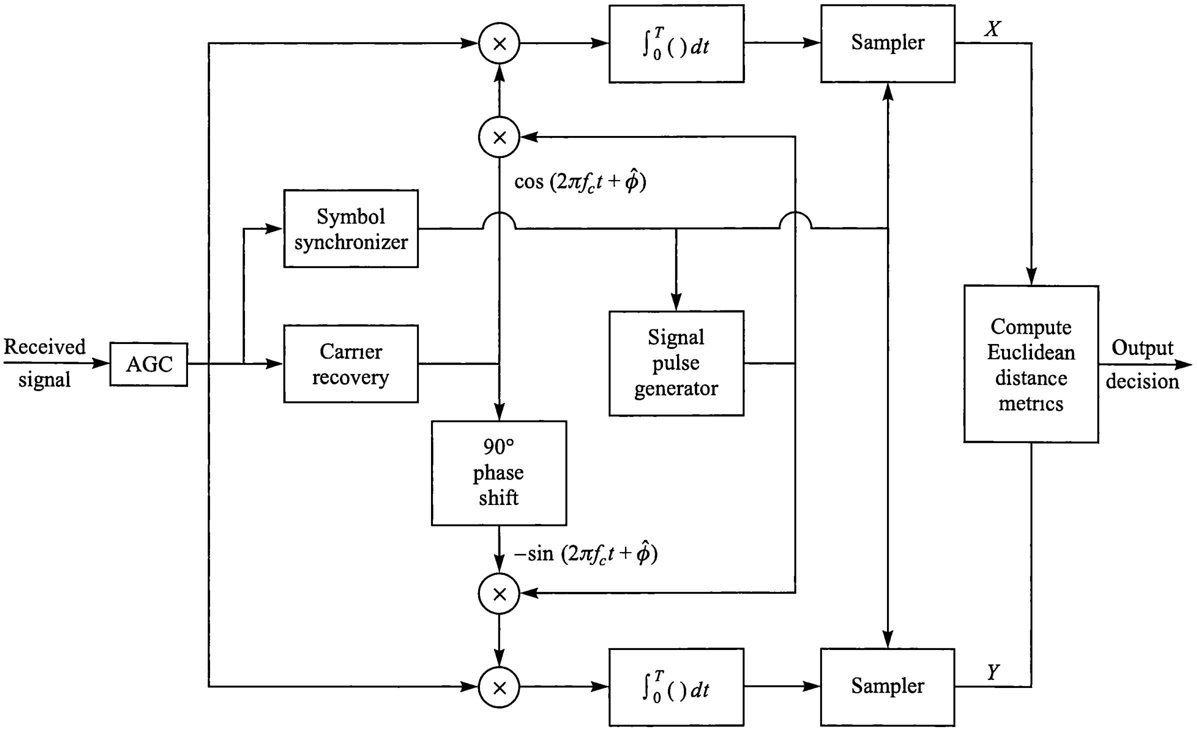

$\int_0^T (\,)dt$
Sampler
X
$\cos(2\pi f_c t+\hat{\phi})$
Symbol synchronizer
Received signal
AGC
Carrier recovery
Signal pulse generator
Compute Euclidean distance metrics
Output decision
90° phase shift
$-\sin(2\pi f_c t+\hat{\phi})$
$\int_0^T (\,)dt$
Sampler
Y

FIGURE 5.1–4
Block diagram of a QAM receiver.

at the input to the demodulator. We observe that the demodulator is similar to a PSK demodulator, in that both generate in-phase and quadrature signal samples (X, Y) for the detector. In the case of QAM, the detector computes the Euclidean distance between the received noise-corrupted signal point and the M possible transmitted points, and selects the signal closest to the received point.

5.2

CARRIER PHASE ESTIMATION

There are two basic approaches for dealing with carrier synchronization at the receiver. One is to multiplex, usually in frequency, a special signal, called a pilot signal, that allows the receiver to extract and, thus, to synchronize its local oscillator to the carrier frequency and phase of the received signal. When an unmodulated carrier component is transmitted along with the information-bearing signal, the receiver employs a phase-locked loop (PLL) to acquire and track the carrier component. The PLL is designed to have a narrow bandwidth so that it is not significantly affected by the presence of frequency components from the information-bearing signal.

The second approach, which appears to be more prevalent in practice, is to derive the carrier phase estimate directly from the modulated signal. This approach has the distinct advantage that the total transmitter power is allocated to the transmission of the information-bearing signal. In our treatment of carrier recovery, we confine our attention to the second approach; hence, we assume that the signal is transmitted via suppressed carrier.

In order to emphasize the importance of extracting an accurate phase estimate, let us consider the effect of a carrier phase error on the demodulation of a double-sideband, suppressed carrier (DSB/SC) signal. To be specific, suppose we have an amplitude-modulated signal of the form

$$s(t) = A(t)\cos(2\pi f_c t + \phi) \tag{5.2–1}$$

If we demodulate the signal by multiplying $s(t)$ with the carrier reference

$$c(t) = \cos(2\pi f_c t + \hat{\phi}) \tag{5.2–2}$$

we obtain

$$c(t)s(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) + \tfrac{1}{2}A(t)\cos(4\pi f_c t + \phi + \hat{\phi})$$

The double-frequency component may be removed by passing the product signal $c(t)s(t)$ through a low-pass filter. This filtering yields the information-bearing signal

$$y(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) \tag{5.2–3}$$

Note that the effect of the phase error $\phi - \hat{\phi}$ is to reduce the signal level in voltage by a factor $\cos(\phi - \hat{\phi})$ and in power by a factor $\cos^2(\phi - \hat{\phi})$. Hence, a phase error of 10° results in a signal power loss of 0.13 dB, and a phase error of 30° results in a signal power loss of 1.25 dB in an amplitude-modulated signal.

The effect of carrier phase errors in QAM and multiphase PSK is much more severe. The QAM and M-PSK signals may be represented as

$$s(t) = A(t)\cos(2\pi f_c t + \phi) - B(t)\sin(2\pi f_c t + \phi) \tag{5.2–4}$$

This signal is demodulated by the two quadrature carriers

$$\begin{aligned} c_i(t) &= \cos(2\pi f_c t + \hat{\phi}) \\ c_q(t) &= -\sin(2\pi f_c t + \hat{\phi}) \end{aligned} \tag{5.2–5}$$

Multiplication of $s(t)$ with $c_i(t)$ followed by low-pass filtering yields the in-phase component

$$y_I(t) = \tfrac{1}{2}A(t)\cos(\phi - \hat{\phi}) - \tfrac{1}{2}B(t)\sin(\phi - \hat{\phi}) \tag{5.2–6}$$

Similarly, multiplication of $s(t)$ by $c_q(t)$ followed by low-pass filtering yields the quadrature component

$$y_Q(t) = \tfrac{1}{2}B(t)\cos(\phi - \hat{\phi}) + \tfrac{1}{2}A(t)\sin(\phi - \hat{\phi}) \tag{5.2–7}$$

The expressions 5.2–6 and 5.2–7 clearly indicate that the phase error in the demodulation of QAM and M-PSK signals has a much more severe effect than in the demodulation of a PAM signal. Not only is there a reduction in the power of the desired signal component by a factor $\cos^2(\phi - \hat{\phi})$, but there is also crosstalk interference from the in-phase and quadrature components. Since the average power levels of $A(t)$ and $B(t)$ are similar, a small phase error causes a large degradation in performance. Hence, the phase accuracy requirements for QAM and multiphase coherent PSK are much higher than for DSB/SC PAM.

5.2–1 Maximum-Likelihood Carrier Phase Estimation

First, we derive the maximum-likelihood carrier phase estimate. For simplicity, we assume that the delay τ is known and, in particular, we set $\tau = 0$. The function to be maximized is the likelihood function given in Equation 5.1–8. With ϕ substituted for $\boldsymbol{\theta}$, this function becomes

$$\begin{aligned} \Lambda(\phi) &= \exp\left\{-\frac{1}{N_0}\int_{T_0}[r(t) - s(t;\phi)]^2\,dt\right\} \\ &= \exp\left\{-\frac{1}{N_0}\int_{T_0} r^2(t)\,dt + \frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt - \frac{1}{N_0}\int_{T_0} s^2(t;\phi)\,dt\right\} \end{aligned} \tag{5.2–8}$$

Note that the first term of the exponential factor does not involve the signal parameter ϕ. The third term, which contains the integral of $s^2(t;\phi)$, is a constant equal to the signal energy over the observation interval T_0 for any value of ϕ. Only the second term, which involves the cross correlation of the received signal $r(t)$ with the signal $s(t;\phi)$, depends on the choice of ϕ. Therefore, the likelihood function $\Lambda(\phi)$ may be expressed as

$$\Lambda(\phi) = C \exp\left[\frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt\right] \tag{5.2–9}$$

where C is a constant independent of ϕ.

The ML estimate $\hat{\phi}_{\mathrm{ML}}$ is the value of ϕ that maximizes $\Lambda(\phi)$ in Equation 5.2–9. Equivalently, the value $\hat{\phi}_{\mathrm{ML}}$ also maximizes the logarithm of $\Lambda(\phi)$, i.e., the log-likelihood function

$$\Lambda_L(\phi) = \frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt \tag{5.2–10}$$

Note that in defining $\Lambda_L(\phi)$ we have ignored the constant term $\ln C$.

EXAMPLE 5.2–1. As an example of the optimization to determine the carrier phase, let us consider the transmission of the unmodulated carrier $A\cos 2\pi f_c t$. The received signal is

$$r(t) = A\cos(2\pi f_c t + \phi) + n(t)$$

where ϕ is the unknown phase. We seek the value ϕ, say $\hat{\phi}_{\mathrm{ML}}$, that maximizes

$$\Lambda_L(\phi) = \frac{2A}{N_0}\int_{T_0} r(t)\cos(2\pi f_c t + \phi)\,dt$$

A necessary condition for a maximum is that

$$\frac{d\Lambda_L(\phi)}{d\phi} = 0$$

This condition yields

$$\int_{T_0} r(t)\sin(2\pi f_c t + \hat{\phi}_{\mathrm{ML}})\,dt = 0 \tag{5.2–11}$$

or, equivalently,

$$\hat{\phi}_{\mathrm{ML}} = -\tan^{-1}\left[\int_{T_0} r(t)\sin 2\pi f_c t\,dt \bigg/ \int_{T_0} r(t)\cos 2\pi f_c t\,dt\right] \tag{5.2–12}$$

We observe that the optimality condition given by Equation 5.2–11 implies the use of a loop to extract the estimate as illustrated in Figure 5.2–1. The loop filter is an integrator whose bandwidth is proportional to the reciprocal of the integration interval T_0. On the other hand, Equation 5.2–12 implies an implementation that uses quadrature carriers to cross-correlate with $r(t)$. Then $\hat{\phi}_{\mathrm{ML}}$ is the inverse tangent of the ratio of these two correlator outputs, as shown in Figure 5.2–2. Note that this estimation scheme yields $\hat{\phi}_{\mathrm{ML}}$ explicitly.

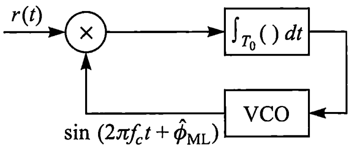


FIGURE 5.2–1
A PLL for obtaining the ML estimate of the phase of an unmodulated carrier.

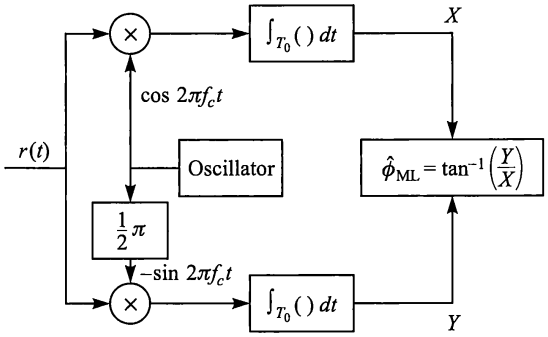


FIGURE 5.2–2
A (one-shot) ML estimate of the phase of an unmodulated carrier.

This example clearly demonstrates that the PLL provides the ML estimate of the phase of an unmodulated carrier.

5.2–2 The Phase-Locked Loop

The PLL basically consists of a multiplier, a loop filter, and a voltage-controlled oscillator (VCO), as shown in Figure 5.2–3. If we assume that the input to the PLL is the sinusoid $\cos(2\pi f_c t + \phi)$ and the output of the VCO is $\sin(2\pi f_c t + \hat{\phi})$, where $\hat{\phi}$ represents the estimate of ϕ, the product of these two signals is

$$\begin{aligned} e(t) &= \cos(2\pi f_c t + \phi)\sin(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}\sin(\hat{\phi} - \phi) + \tfrac{1}{2}\sin(4\pi f_c t + \phi + \hat{\phi}) \end{aligned} \tag{5.2–13}$$

The loop filter is a low-pass filter that responds only to the low-frequency component $\frac{1}{2}\sin(\hat{\phi} - \phi)$ and removes the component at $2f_c$. This filter is usually selected to have the relatively simple transfer function

$$G(s) = \frac{1 + \tau_2 s}{1 + \tau_1 s} \tag{5.2–14}$$

where τ_1 and τ_2 are design parameters ($\tau_1 \gg \tau_2$) that control the bandwidth of the loop. A higher-order filter that contains additional poles may be used if necessary to obtain a better loop response.

The output of the loop filter provides the control voltage $v(t)$ for the VCO. The VCO is basically a sinusoidal signal generator with an instantaneous phase given by

$$2\pi f_c t + \hat{\phi}(t) = 2\pi f_c t + K \int_{-\infty}^{t} v(\tau)\,d\tau \tag{5.2–15}$$

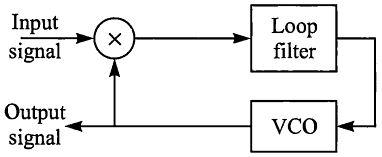


FIGURE 5.2–3
Basic elements of a phase-locked loop (PLL).

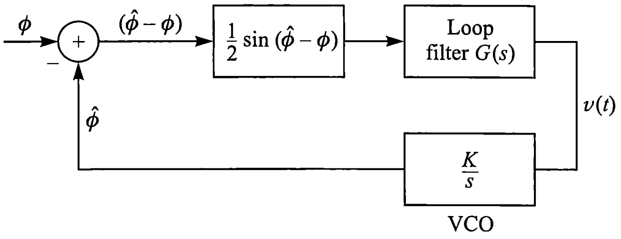


FIGURE 5.2–4
Model of phase-locked loop.

where K is a gain constant in rad/V. Hence,

$$\hat{\phi}(t) = K \int_{-\infty}^{t} v(\tau)\, d\tau \tag{5.2–16}$$

By neglecting the double-frequency term resulting from the multiplication of the input signal with the output of the VCO, we may reduce the PLL into the equivalent closed-loop system model shown in Figure 5.2–4. The sine function of the phase difference $\hat{\phi} - \phi$ makes this system non-linear, and, as a consequence, the analysis of its performance in the presence of noise is somewhat involved, but, nevertheless, it is mathematically tractable for some simple loop filters.

In normal operation when the loop is tracking the phase of the incoming carrier, the phase error $\hat{\phi} - \phi$ is small and, hence,

$$\sin(\hat{\phi} - \phi) \approx \hat{\phi} - \phi \tag{5.2–17}$$

With this approximation, the PLL becomes linear and is characterized by the closed-loop transfer function

$$H(s) = \frac{KG(s)/s}{1 + KG(s)/s} \tag{5.2–18}$$

where the factor of $\frac{1}{2}$ has been absorbed into the gain parameter K. By substituting from Equation 5.2–14 for $G(s)$ into Equation 5.2–18, we obtain

$$H(s) = \frac{1 + \tau_2 s}{1 + (\tau_2 + 1/K)s + (\tau_1/K)s^2} \tag{5.2–19}$$

Hence, the closed-loop system for the linearized PLL is second-order when $G(s)$ is given by Equation 5.2–14. The parameter τ_2 controls the position of the zero, while K and τ_1 are used to control the position of the closed-loop system poles. It is customary to express the denominator of $H(s)$ in the standard form

$$D(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{5.2–20}$$

where ζ is called the *loop damping factor* and ω_n is the natural frequency of the loop. In terms of the loop parameters, $\omega_n = \sqrt{K/\tau_1}$, and $\zeta = \omega_n(\tau_2 + 1/K)/2$, the closed-loop transfer function becomes

$$H(s) = \frac{(2\zeta\omega_n - \omega_n^2/K)s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.2–21}$$

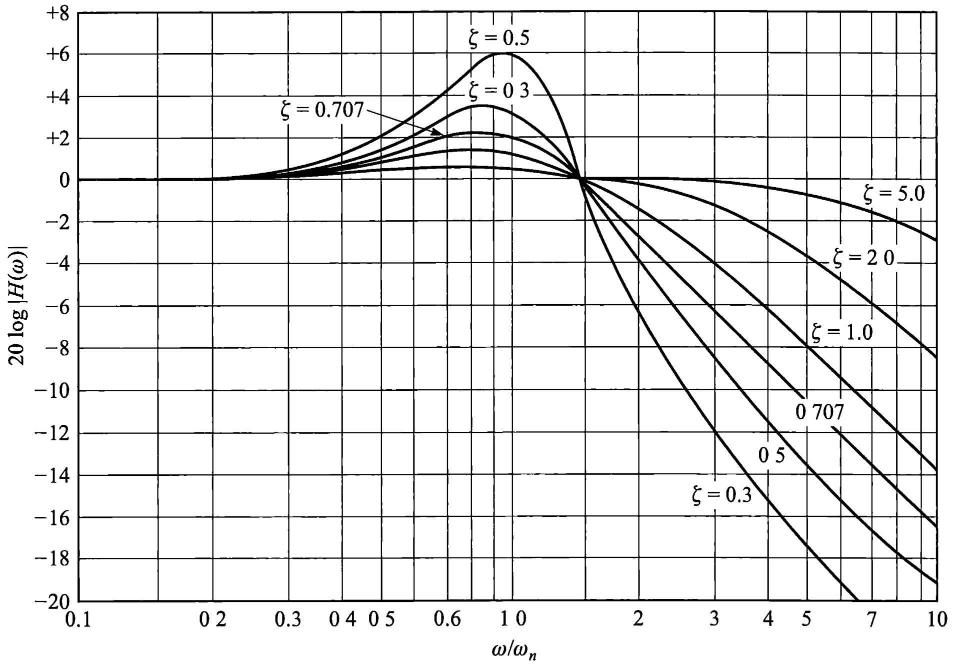


FIGURE 5.2–5
Frequency response of a second-order loop. [*From* Phaselock Techniques, *2nd edition, by F. M. Gardner, © 1979 by John Wiley and Sons, Inc. Reprinted with permission of the publisher.*]

The (one-sided) noise-equivalent bandwidth (see Problem 2.52) of the loop is

$$B_{\mathrm{eq}} = \frac{\tau_2^2(1/\tau_2^2 + K/\tau_1)}{4(\tau_2 + 1/K)} = \frac{1 + (\tau_2\omega_n)^2}{8\zeta/\omega_n} \tag{5.2–22}$$

The magnitude response 20 log $|H(\omega)|$ as a function of the normalized frequency ω/ω_n is illustrated in Figure 5.2–5, with the damping factor ζ as a parameter and $\tau_1 \gg 1$. Note that $\zeta = 1$ results in a critically damped loop response, $\zeta < 1$ produces an underdamped response, and $\zeta > 1$ yields an overdamped response.

In practice, the selection of the bandwidth of the PLL involves a tradeoff between speed of response and noise in the phase estimate, which is the topic considered below. On the one hand, it is desirable to select the bandwidth of the loop to be sufficiently wide to track any time variations in the phase of the received carrier. On the other hand, a wideband PLL allows more noise to pass into the loop, which corrupts the phase estimate. Below, we assess the effects of noise in the quality of the phase estimate.

5.2–3 Effect of Additive Noise on the Phase Estimate

In order to evaluate the effects of noise on the estimate of the carrier phase, let us assume that the noise at the input to the PLL is narrowband. For this analysis, we assume that the PLL is tracking a sinusoidal signal of the form

$$s(t) = A_c \cos[2\pi f_c t + \phi(t)] \tag{5.2–23}$$

that is corrupted by the additive narrowband noise

$$n(t) = x(t) \cos 2\pi f_c t - y(t) \sin 2\pi f_c t \tag{5.2–24}$$

The in-phase and quadrature components of the noise are assumed to be statistically independent, stationary Gaussian noise processes with (two-sided) power spectral density $\frac{1}{2}N_0$ W/Hz. By using simple trigonometric identities, the noise term in Equation 5.2–24 can be expressed as

$$n(t) = n_i(t) \cos[2\pi f_c t + \phi(t)] - n_q(t) \sin[2\pi f_c t + \phi(t)] \tag{5.2–25}$$

where

$$\begin{aligned} n_i(t) &= x(t) \cos \phi(t) + y(t) \sin \phi(t) \\ n_q(t) &= -x(t) \sin \phi(t) + y(t) \cos \phi(t) \end{aligned} \tag{5.2–26}$$

We note that

$$n_i(t) + jn_q(t) = [x(t) + jy(t)]e^{-j\phi(t)}$$

so that the quadrature components $n_i(t)$ and $n_q(t)$ have exactly the same statistical characteristics as $x(t)$ and $y(t)$.

If $s(t) + n(t)$ is multiplied by the output of the VCO and the double-frequency terms are neglected, the input to the loop filter is the noise-corrupted signal

$$\begin{aligned} e(t) &= A_c \sin \Delta\phi + n_i(t) \sin \Delta\phi - n_q(t) \cos \Delta\phi \\ &= A_c \sin \Delta\phi + n_1(t) \end{aligned} \tag{5.2–27}$$

where, by definition $\Delta\phi = \hat{\phi} - \phi$ is the phase error. Thus, we have the equivalent model for the PLL with additive noise as shown in Figure 5.2–6.

When the power $P_c = \frac{1}{2}A_c^2$ of the incoming signal is much larger than the noise power, we may linearize the PLL and, thus, easily determine the effect of the additive noise on the quality of the estimate $\hat{\phi}$. Under these conditions, the model for the

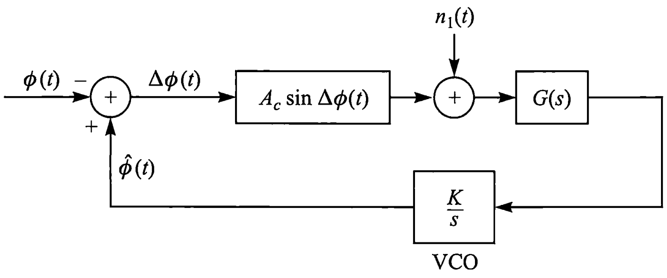


FIGURE 5.2–6
Equivalent PLL model with additive noise.

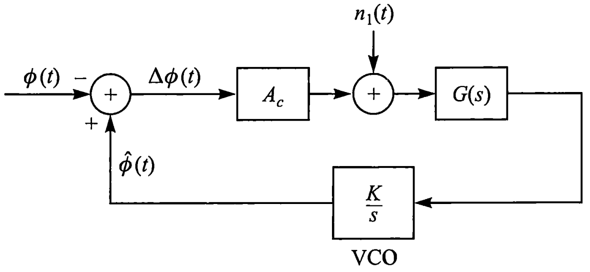


FIGURE 5.2–7
Linearized PLL model with additive noise.

linearized PLL with additive noise is illustrated in Figure 5.2–7. Note that the gain parameter A_c may be normalized to unity, provided that the noise terms are scaled by $1/A_c$, i.e., the noise terms become

$$n_2(t) = \frac{n_i(t)}{A_c} \sin \Delta\phi - \frac{n_q(t)}{A_c} \cos \Delta\phi \tag{5.2–28}$$

The noise term $n_2(t)$ is zero-mean Gaussian with a power spectral density $N_0/2A_c^2$. Since the noise $n_2(t)$ is additive at the input to the loop, the variance of the phase error $\Delta\phi$, which is also the variance of the VCO output phase, is

$$\begin{aligned} \sigma_{\hat{\phi}}^2 &= \frac{N_0}{2A_c^2} \int_{-\infty}^{\infty} |H(f)|^2 \, df \\ &= \frac{N_0}{A_c^2} \int_{0}^{\infty} |H(f)|^2 \, df \\ &= \frac{N_0 B_{\text{eq}}}{A_c^2} \end{aligned} \tag{5.2–29}$$

where B_{eq} is the (one-sided) equivalent noise bandwidth of the loop, given in Equation 5.2–22. Note that $\sigma_{\hat{\phi}}^2$ is simply the ratio of total noise power within the bandwidth of the PLL divided by the signal power. Hence,

$$\sigma_{\hat{\phi}}^2 = \frac{1}{\gamma_L} \tag{5.2–30}$$

where γ_L is defined as the signal-to-noise ratio

$$\text{SNR} \equiv \gamma_L = \frac{A_c^2}{N_0 B_{\text{eq}}} \tag{5.2–31}$$

The expression for the variance $\sigma_{\hat{\phi}}^2$ of the VCO phase error applies to the case where the SNR is sufficiently high that the linear model for the PLL applies. An exact analysis based on the non-linear PLL is mathematically tractable when $G(s) = 1$, which results in a first-order loop. In this case, the probability density function for the phase error may be derived (see Viterbi, 1966) and has the form

$$p(\Delta\phi) = \frac{\exp(\gamma_L \cos \Delta\phi)}{2\pi I_0(\gamma_L)} \tag{5.2–32}$$

where γ_L is the SNR given by Equation 5.2–31 with B_{eq} being the appropriate noise bandwidth of the first-order loop, and $I_0(\cdot)$ is the modified Bessel function of order zero.

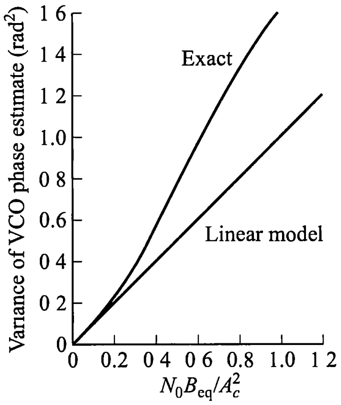


FIGURE 5.2–8
Comparison of VCO phase variance for exact and approximate (linear model) first-order PLL. [*From* Principles of Coherent Communication, *by A. J. Viterbi; ©1966 by McGraw-Hill Book Company. Reprinted with permission of the publisher.*]

From the expression for $p(\Delta\phi)$, we may obtain the exact value of the variance for the phase error on a first-order PLL. This is plotted in Figure 5.2–8 as a function of $1/\gamma_L$. Also shown for comparison is the result obtained with the linearized PLL model. Note that the variance for the linear model is close to the exact variance for $\gamma_L > 3$. Hence, the linear model is adequate for practical purposes.

Approximate analyses of the statistical characteristics of the phase error for the nonlinear PLL have also been performed. Of particular importance is the transient behavior of the PLL during initial acquisition. Another important problem is the behavior of PLL at low SNR. It is known, for example, that when the SNR at the input to the PLL drops below a certain value, there is a rapid deterioration in the performance of the PLL. The loop begins to lose lock and an impulsive type of noise, characterized as clicks, is generated which degrades the performance of the loop. Results on these topics can be found in the texts by Viterbi (1966), Lindsey (1972), Lindsey and Simon (1973), and Gardner (1979), and in the survey papers by Gupta (1975) and Lindsey and Chie (1981).

Up to this point, we have considered carrier phase estimation when the carrier signal is unmodulated. Below, we consider carrier phase recovery when the signal carries information.

5.2–4 Decision-Directed Loops

A problem arises in maximizing either Equation 5.2–9 or 5.2–10 when the signal $s(t;\,\phi)$ carries the information sequence $\{I_n\}$. In this case we can adopt one of two approaches: either we assume that $\{I_n\}$ is known or we treat $\{I_n\}$ as a random sequence and average over its statistics.

In decision-directed parameter estimation, we assume that the information sequence $\{I_n\}$ over the observation interval has been estimated and, in the absence of demodulation errors, $\tilde{I}_n = I_n$, where $\tilde{I}_n$ denotes the detected value of the information I_n. In this case $s(t;\,\phi)$ is completely known except for the carrier phase.

To be specific, let us consider the decision-directed phase estimate for the class of linear modulation techniques for which the received *equivalent low-pass signal* may be expressed as

$$r_l(t) = e^{-j\phi}\sum_n I_n g(t-nT) + z(t) = s_l(t)e^{-j\phi} + z(t) \tag{5.2–33}$$

where $s_l(t)$ is a known signal if the sequence $\{I_n\}$ is assumed known. The likelihood function and corresponding log-likelihood function for the equivalent low-pass signal are

$$\Lambda(\phi) = C \exp\left\{\mathrm{Re}\left[\frac{1}{N_0}\int_{T_0} r_l(t)s_l^*(t)e^{j\phi}dt]\right]\right\} \tag{5.2–34}$$

$$\Lambda_L(\phi) = \mathrm{Re}\left\{\left[\frac{1}{N_0}\int_{T_0} r_l(t)s_l^*(t)\,dt\right]e^{j\phi}\right\} \tag{5.2–35}$$

If we substitute for $s_l(t)$ in Equation 5.2–35 and assume that the observation interval $T_0 = KT$, where K is a positive integer, we obtain

$$\begin{aligned}\Lambda_L(\phi) &= \mathrm{Re}\left\{e^{j\phi}\frac{1}{N_0}\sum_{n=0}^{K-1} I_n^* \int_{nT}^{(n+1)T} r_l(t)g^*(t-nT)\,dt\right\} \\ &= \mathrm{Re}\left\{e^{j\phi}\frac{1}{N_0}\sum_{n=0}^{K-1} I_n^* y_n\right\}\end{aligned} \tag{5.2–36}$$

where, by definition

$$y_n = \int_{nT}^{(n+1)T} r_l(t)g^*(t-nT)\,dt \tag{5.2–37}$$

Note that y_n is the output of the matched filter in the nth signal interval. The ML estimate of ϕ is easily found from Equation 5.2–36 by differentiating the log-likelihood

$$\Lambda_L(\phi) = \mathrm{Re}\left(\frac{1}{N_0}\sum_{n=0}^{K-1} I_n^* y_n\right)\cos\phi - \mathrm{Im}\left(\frac{1}{N_0}\sum_{n=0}^{K-1} I_n^* y_n\right)\sin\phi$$

with respect to ϕ and setting the derivative equal to zero. Thus, we obtain

$$\hat{\phi}_{ML} = -\tan^{-1}\left[\mathrm{Im}\left(\sum_{n=0}^{K-1} I_n^* y_n\right)\Big/\mathrm{Re}\left(\sum_{n=0}^{K-1} I_n^* y_n\right)\right] \tag{5.2–38}$$

We call $\hat{\phi}_{\mathrm{ML}}$ in Equation 5.2–38 the *decision-directed* (or *decision-feedback) carrier phase estimate.* It is easily shown (Problem 5.10) that the mean value of $\hat{\phi}_{\mathrm{ML}}$ is ϕ, so that the estimate is unbiased. Furthermore, the PDF of $\hat{\phi}_{\mathrm{ML}}$ can be obtained (Problem 5.11) by using the procedure described in Section 4.3–2.

The block diagram of a double-sideband PAM signal receiver that incorporates the decision-directed carrier phase estimate given by Equation 5.2–38 is illustrated in Figure 5.2–9.

Another implementation of the PAM receiver that employs a decision-feedback PLL (DFPLL) for carrier phase estimation is shown in Figure 5.2–10. The received double-sideband PAM signal is given by $A(t)\cos(2\pi f_c t + \phi)$, where $A(t) = A_m g(t)$ and $g(t)$ is assumed to be a rectangular pulse of duration T. This received signal is multiplied by the quadrature carriers $c_i(t)$ and $c_q(t)$, as given by Equation 5.2–5, which

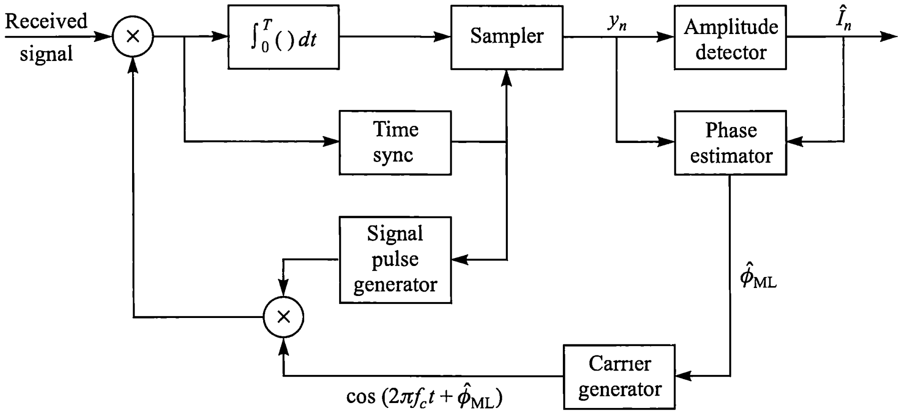

Received signal
×
$\int_0^T ()\,dt$
Sampler
y_n
Amplitude detector
$\hat{I}_n$
Time sync
Phase estimator
Signal pulse generator
$\hat{\phi}_{ML}$
×
Carrier generator
$\cos(2\pi f_c t + \hat{\phi}_{ML})$

FIGURE 5.2–9
Block diagram of double-sideband PAM signal receiver with decision-directed carrier phase estimation.

are derived from the VCO. The product signal

$$
\begin{aligned}
r(t)\cos(2\pi f_c t + \hat{\phi}) = {} & \tfrac{1}{2}[A(t) + n_i(t)]\cos\Delta\phi \\
& - \tfrac{1}{2}n_q(t)\sin\Delta\phi + \text{double-frequency terms}
\end{aligned}
\tag{5.2–39}
$$

is used to recover the information carried by $A(t)$. The detector makes a decision on the symbol that is received every T seconds. Thus, in the absence of decision errors, it reconstructs $A(t)$ free of any noise. This reconstructed signal is used to multiply the product of the second quadrature multiplier, which has been delayed by T seconds to allow the demodulator to reach a decision. Thus, the input to the loop filter in the absence of decision errors is the error signal

$$
\begin{aligned}
e(t) &= \tfrac{1}{2}A(t)\{[A(t) + n_i(t)]\sin\Delta\phi - n_q(t)\cos\Delta\phi\} \\
&\quad + \text{double-frequency terms} \\
&= \tfrac{1}{2}A^2(t)\sin\Delta\phi + \tfrac{1}{2}A(t)[n_i(t)\sin\Delta\phi - n_q(t)\cos\Delta\phi] \\
&\quad + \text{double-frequency terms}
\end{aligned}
\tag{5.2–40}
$$

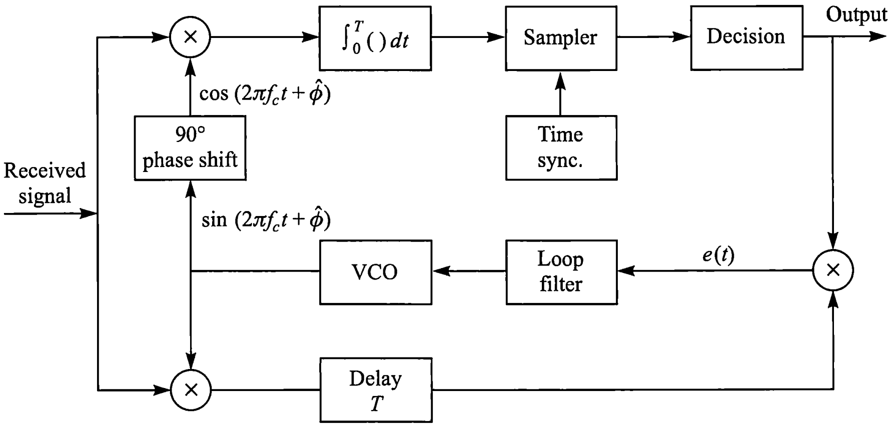
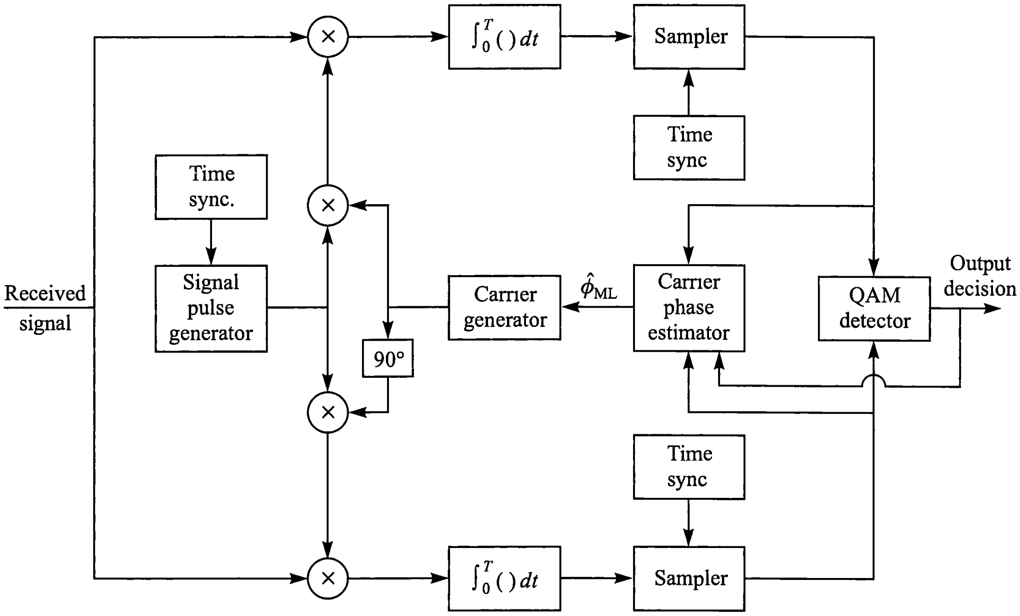

×
$\int_0^T ()\,dt$
Sampler
Decision
Output
$\cos(2\pi f_c t + \hat{\phi})$
90° phase shift
Time sync.
Received signal
$\sin(2\pi f_c t + \hat{\phi})$
VCO
Loop filter
$e(t)$
×
×
Delay T

FIGURE 5.2–10
Carrier recovery with a decision-feedback PLL.



$\int_0^T (\,)dt$
Sampler
Time sync
Time sync.
Signal pulse generator
Carrier generator
$\hat{\phi}_{\mathrm{ML}}$
Carrier phase estimator
QAM detector
Output decision
Received signal
90°
$\int_0^T (\,)dt$
Time sync
Sampler

FIGURE 5.2–11
Block diagram of QAM signal receiver with decision-directed carrier phase estimation.

The loop filter is low-pass and, hence, it rejects the double-frequency term in $e(t)$. The desired component is $A^2(t)\sin\Delta\phi$, which contains the phase error for driving the loop.

The ML estimate in Equation 5.2–38 is also appropriate for QAM. The block diagram of a QAM receiver that incorporates the decision-directed carrier phase estimate is shown in Figure 5.2–11.

In the case of M-ary PSK, the DFPLL has the configuration shown in Figure 5.2–12. The received signal is demodulated to yield the phase estimate

$$\hat{\theta}_m = \frac{2\pi}{M}(m-1)$$

which, in the absence of a decision error, is the transmitted signal phase θ_m. The two outputs of the quadrature multipliers are delayed by the symbol duration T and multiplied by $\cos\theta_m$ and $\sin\theta_m$ to yield

$$\begin{aligned}
&r(t)\cos(2\pi f_c t + \hat{\phi})\sin\theta_m \\
&\quad = \tfrac{1}{2}[A\cos\theta_m + n_i(t)]\sin\theta_m\cos(\phi - \hat{\phi}) \\
&\quad\quad - \tfrac{1}{2}[A\sin\theta_m + n_q(t)]\sin\theta_m\sin(\phi - \hat{\phi}) \\
&\quad\quad + \text{double-frequency terms} \qquad\qquad (5.2\text{–}41) \\
&r(t)\sin(2\pi f_c t + \hat{\phi})\cos\theta_m \\
&\quad = -\tfrac{1}{2}[A\cos\theta_m + n_i(t)]\cos\theta_m\sin(\phi - \hat{\phi}) \\
&\quad\quad - \tfrac{1}{2}[A\sin\theta_m + n_q(t)]\cos\theta_m\cos(\phi - \hat{\phi}) \\
&\quad\quad + \text{double-frequency terms}
\end{aligned}$$

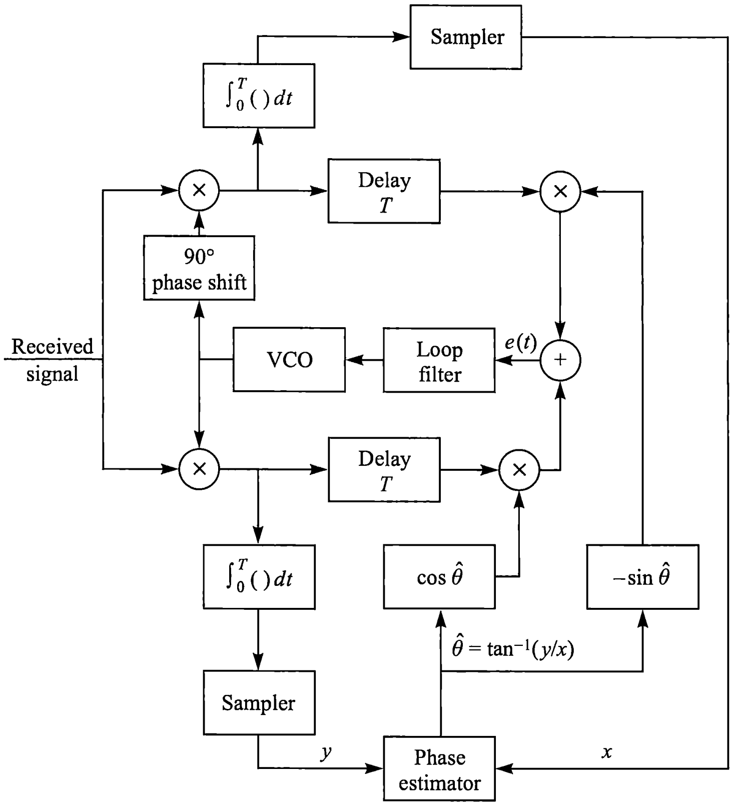

Sampler
$\int_0^T (\,)dt$
Delay
T
$90°$ phase shift
Received signal
VCO
Loop filter
$e(t)$
+
Delay
T
$\int_0^T (\,)dt$
$\cos\hat{\theta}$
$-\sin\hat{\theta}$
$\hat{\theta} = \tan^{-1}(y/x)$
Sampler
y
Phase estimator
x

FIGURE 5.2–12
Carrier recovery for M-ary PSK using a decision-feedback PLL.

The two signals are added to generate the error signal

$$\begin{aligned} e(t) &= -\tfrac{1}{2}A\sin(\phi-\hat{\phi}) + \tfrac{1}{2}n_i(t)\sin(\phi-\hat{\phi}-\theta_m) \\ &\quad + \tfrac{1}{2}n_q(t)\cos(\phi-\hat{\phi}-\theta_m) + \text{double-frequency terms} \end{aligned} \tag{5.2–42}$$

This error signal is the input to the loop filter that provides the control signal for the VCO.

We observe that the two quadrature noise components in Equation 5.2–42 appear as additive terms. There is no term involving a product of two noise components as in an Mth-power law device, described in the next section. Consequently, there is no additional power loss associated with the decision-feedback PLL.

This M-phase tracking loop has a phase ambiguity of $360°/M$, necessitating the need to differentially encode the information sequence prior to transmission and differentially decode the received sequence after demodulation to recover the information.

The ML estimate in Equation 5.2–38 is also appropriate for QAM. The ML estimate for offset QPSK is also easily obtained (Problem 5.12) by maximizing the log-likelihood function in Equation 5.2–35, with $s_l(t)$ given as

$$s_l(t) = \textstyle\sum_n I_n g(t-nT) + j\sum_n J_n g(t-nT-\tfrac{1}{2}T) \tag{5.2–43}$$

where $I_n = \pm 1$ and $J_n = \pm 1$.

Finally, we should also mention that carrier phase recovery for CPM signals can also be accomplished in a decision-directed manner by use of a PLL. From the optimum demodulator for CPM signals, which is described in Section 4.3, we can generate an error signal that is filtered in a loop filter whose output drives a PLL. Alternatively, we may exploit the linear representation of CPM signals and, thus, employ a generalization of the carrier phase estimator given by Equation 5.2–38, in which the cross correlation of the received signal is performed with each of the pulses in the linear representation. A comprehensive description of carrier phase recover techniques for CPM is given in the book by Mengali and D'Andrea (1997).

5.2–5 Non-Decision-Directed Loops

Instead of using a decision-directed scheme to obtain the phase estimate, we may treat the data as random variables and simply average $\Lambda(\phi)$ over these random variables prior to maximization. In order to carry out this integration, we may use either the actual probability distribution function of the data, if it is known, or, perhaps, we may assume some probability distribution that might be a reasonable approximation to the true distribution. The following example illustrates the first approach.

EXAMPLE 5.2–2. Suppose the real signal $s(t)$ carries binary modulation. Then, in a signal interval, we have

$$s(t) = A\cos 2\pi f_c t, \qquad 0 \le t \le T$$

where $A = \pm 1$ with equal probability. Clearly, the PDF of A is given as

$$p(A) = \tfrac{1}{2}\delta(A-1) + \tfrac{1}{2}\delta(A+1)$$

Now, the likelihood function $\Lambda(\phi)$ given by Equation 5.2–9 may be considered as conditional on a given value of A and must be averaged over the two values. Thus,

$$\begin{aligned}\bar{\Lambda}(\phi) &= \int_{-\infty}^{\infty} \Lambda(\phi)p(A)\,dA \\ &= \tfrac{1}{2}\exp\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right] \\ &\quad + \tfrac{1}{2}\exp\left[-\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right] \\ &= \cosh\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right]\end{aligned}$$

and the corresponding log-likelihood function is

$$\bar{\Lambda}_L(\phi) = \ln\cosh\left[\frac{2}{N_0}\int_0^T r(t)\cos(2\pi f_c t + \phi)\,dt\right] \tag{5.2–44}$$

If we differentiate $\bar{\Lambda}_L(\phi)$ and set the derivative equal to zero, we obtain the ML estimate for the non-decision-directed estimate. Unfortunately, the functional relationship in

Equation 5.2–44 is highly non-linear and, hence, an exact solution is difficult to obtain. On the other hand, approximations are possible. In particular,

$$\ln\cosh x = \begin{cases} \frac{1}{2}x^2 & (|x| \ll 1) \\ |x| & (|x| \gg 1) \end{cases} \tag{5.2–45}$$

With these approximations, the solution for ϕ becomes tractable.

In this example, we averaged over the two possible values of the information symbol. When the information symbols are M-valued, where M is large, the averaging operation yields highly non-linear functions of the parameter to be estimated. In such a case, we may simplify the problem by assuming that the information symbols are continuous random variables. For examples, we may assume that the symbols are zero-mean Gaussian. The following example illustrates this approximation and the resulting form for the average likelihood function.

EXAMPLE 5.2–3. Let us consider the same signal as in Example 5.2–2, but now we assume that the amplitude A is zero-mean Gaussian with unit variance. Thus,

$$p(A) = \frac{1}{\sqrt{2\pi}} e^{-A^2/2}$$

If we average $\Lambda(\phi)$ over the assumed PDF of A, we obtain the average likelihood $\bar{\Lambda}(\phi)$ in the form

$$\bar{\Lambda}(\phi) = C \exp\left\{ \left[\frac{2}{N_0} \int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt \right]^2 \right\} \tag{5.2–46}$$

and the corresponding log-likelihood as

$$\bar{\Lambda}_L(\phi) = \left[\frac{2}{N_0} \int_0^T r(t)\cos(2\pi f_c t + \phi)\, dt \right]^2 \tag{5.2–47}$$

We can obtain the ML estimate of ϕ by differentiating $\bar{\Lambda}_L(\phi)$ and setting the derivative to zero.

It is interesting to note that the log-likelihood function is quadratic under the Gaussian assumption and that it is approximately quadratic, as indicated in Equation 5.2–45 for small values of the cross correlation of $r(t)$ with $s(t;\phi)$. In other words, if the cross correlation over a single interval is small, the Gaussian assumption for the distribution of the information symbols yields a good approximation to the log-likelihood function.

In view of these results, we may use the Gaussian approximation on all the symbols in the observation interval $T_0 = KT$. Specifically, we assume that the K information symbols are statistically independent and identically distributed. By averaging the likelihood function $\Lambda(\phi)$ over the Gaussian PDF for each of the K symbols in the interval $T_0 = KT$, we obtain the result

$$\bar{\Lambda}(\phi) = C \exp\left\{ \sum_{n=0}^{K-1} \left[\frac{2}{N_0} \int_{nT}^{(n+1)T} r(t)\cos(2\pi f_c t + \phi)\, dt \right]^2 \right\} \tag{5.2–48}$$

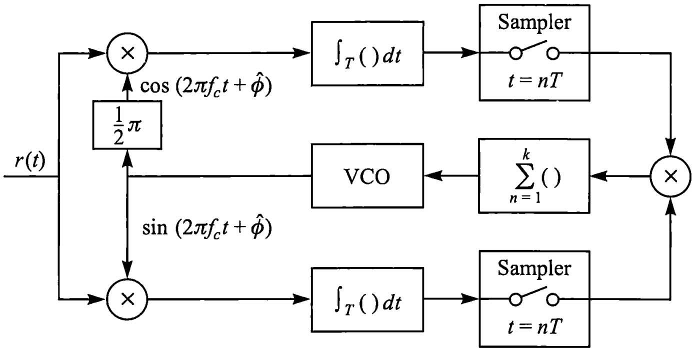


FIGURE 5.2–13

Non-decision-directed PLL for carrier phase estimation of PAM signals.

If we take the logarithm of Equation 5.2–48, differentiate the resulting log-likelihood function, and set the derivative equal to zero, we obtain the condition for the ML estimate as

$$\sum_{n=0}^{K-1}\int_{nT}^{(n+1)T} r(t)\cos(2\pi f_c t+\hat{\phi})\,dt\int_{nT}^{(n+1)T} r(t)\sin(2\pi f_c t+\hat{\phi})\,dt=0 \qquad (5.2\text{–}49)$$

Although this equation can be manipulated further, its present form suggests the tracking loop configuration illustrated in Figure 5.2–13. This loop resembles a Costas loop, which is described below. We note that the multiplication of the two signals from the integrators destroys the sign carried by the information symbols. The summer plays the role of the loop filter. In a tracking loop configuration, the summer may be implemented either as a sliding-window digital filter (summer) or as a low-pass digital filter with exponential weighting of the past data.

In a similar manner, one can derive non-decision-directed ML phase estimates for QAM and M-PSK. The starting point is to average the likelihood function given by Equation 5.2–9 over the statistical characteristics of the data. Here again, we may use the Gaussian approximation (two-dimensional Gaussian for complex-valued information symbols) in averaging over the information sequence.

Squaring loop The squaring loop is a non-decision-directed loop that is widely used in practice to establish the carrier phase of double-sideband suppressed carrier signals such as PAM. To describe its operation, consider the problem of estimating the carrier phase of the digitally modulated PAM signal of the form

$$s(t)=A(t)\cos(2\pi f_c t+\phi) \qquad (5.2\text{–}50)$$

where $A(t)$ carries the digital information. Note that $E[s(t)]=E[A(t)]=0$ when the signal levels are symmetric about zero. Consequently, the average value of $s(t)$ does not produce any phase coherent frequency components at any frequency, including the carrier. One method for generating a carrier from the received signal is to square the signal and, thus, to generate a frequency component at $2f_c$, which can be used to drive a PLL tuned to $2f_c$. This method is illustrated in the block diagram shown in Figure 5.2–14.

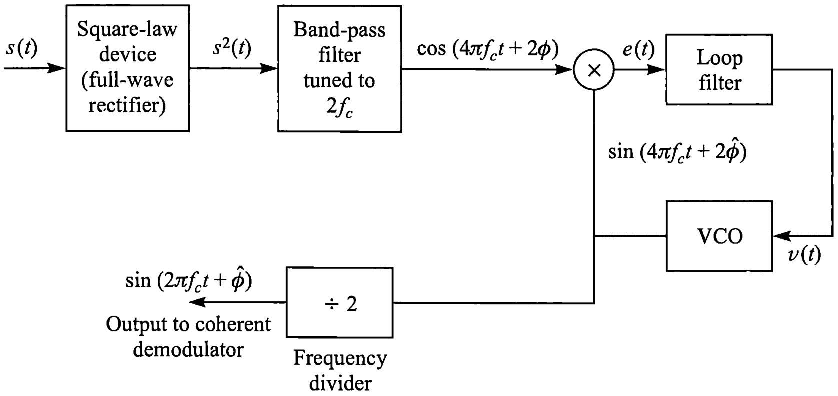


FIGURE 5.2–14
Carrier recover using a square-law device.

The output of the square-law device is

$$\begin{aligned} s^2(t) &= A^2(t)\cos^2(2\pi f_c t + \phi) \\ &= \tfrac{1}{2}A^2(t) + \tfrac{1}{2}A^2(t)\cos(4\pi f_c t + 2\phi) \end{aligned} \tag{5.2–51}$$

Since the modulation is a cyclostationary stochastic process, the expected value of $s^2(t)$ is

$$E[s^2(t)] = \tfrac{1}{2}E[A^2(t)] + \tfrac{1}{2}E[A^2(t)]\cos(4\pi f_c t + 2\phi) \tag{5.2–52}$$

Hence, there is power at the frequency $2f_c$.

If the output of the square-law device is passed through a band-pass filter tuned to the double-frequency term in Equation 5.2–51, the mean value of the filter is a sinusoid with frequency $2f_c$, phase 2ϕ, and amplitude $\frac{1}{2}E[A^2(t)]H(2f_c)$, where $H(2f_c)$ is the gain of the filter at $f = 2f_c$. Thus, the square-law device has produced a periodic component from the input signal $s(t)$. In effect, the squaring of $s(t)$ has removed the sign information contained in $A(t)$ and, thus, has resulted in phase-coherent frequency components at twice the carrier. The filtered frequency component at $2f_c$ is then used to drive the PLL.

The squaring operation leads to a noise enhancement that increases the noise power level at the input to the PLL and results in an increase in the variance of the phase error.

To elaborate on this point, let the input to the squarer be $s(t) + n(t)$, where $s(t)$ is given by Equation 5.2–50 and $n(t)$ represents the band-pass additive Gaussian noise process. By squaring $s(t) + n(t)$, we obtain

$$y(t) = s^2(t) + 2s(t)n(t) + n^2(t) \tag{5.2–53}$$

where $s^2(t)$ is the desired signal component and the other two components are the signal × noise and noise × noise terms. By computing the autocorrelation functions and power density spectra of these two noise components, one can easily show that both components have spectral power in the frequency band centered at $2f_c$. Consequently, the band-pass filter with bandwidth B_{bp} centered at $2f_c$, which produces the desired sinusoidal signal component that drives the PLL, also passes noise due to these two terms.

Since the bandwidth of the loop is designed to be significantly smaller than the bandwidth B_{bp} of the band-pass filter, the total noise spectrum at the input to the PLL may be approximated as a constant within the loop bandwidth. This approximation allows us to obtain a simple expression for the variance of the phase error as

$$\sigma_{\hat{\phi}}^2 = \frac{1}{\gamma_L S_L} \tag{5.2--54}$$

where S_L is called the squaring loss and is given by

$$S_L = \left(1 + \frac{B_{\text{bp}}/2B_{\text{eq}}}{\gamma_L}\right)^{-1} \tag{5.2–55}$$

Since $S_L < 1$, S_L^{-1} represents the increase in the variance of the phase error caused by the added noise (noise × noise terms) that results from the squarer. Note, for example, that when $\gamma_L = B_{\text{bp}}/2B_{\text{eq}}$, the loss is 3 dB.

Finally, we observe that the output of the VCO from the squaring loop must be frequency-divided by 2 to generate the phase-locked carrier for signal demodulation. It should be noted that the output of the frequency divider has a phase ambiguity of 180° relative to the phase of the received signal. For this reason, the data must be differentially encoded prior to transmission and differentially decoded at the receiver.

Costas loop Another method for generating a properly phased carrier for a double-sideband suppressed carrier signal is illustrated by the block diagram shown in Figure 5.2–15. This scheme was developed by Costas (1956) and is called the *Costas loop*. The received signal is multiplied by $\cos(2\pi f_c t + \hat{\phi})$ and $\sin(2\pi f_c t + \hat{\phi})$, which are outputs from the VCO. The two products are

$$\begin{aligned} y_c(t) &= [s(t) + n(t)]\cos(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}[A(t) + n_i(t)]\cos\Delta\phi + \tfrac{1}{2}n_q(t)\sin\Delta\phi \\ &\quad + \text{double-frequency terms} \\ y_s(t) &= [s(t) + n(t)]\sin(2\pi f_c t + \hat{\phi}) \\ &= \tfrac{1}{2}[A(t) + n_i(t)]\sin\Delta\phi - \tfrac{1}{2}n_q(t)\cos\Delta\phi \\ &\quad + \text{double-frequency terms} \end{aligned} \tag{5.2–56}$$

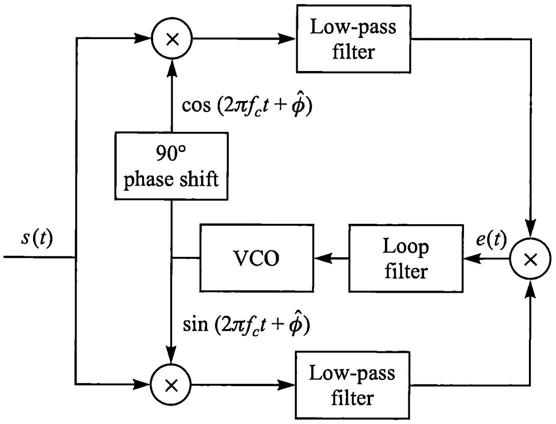


FIGURE 5.2–15
Block diagram of Costas loop.

where the phase error $\Delta\phi = \hat{\phi} - \phi$. The double-frequency terms are eliminated by the low-pass filters following the multiplications.

An error signal is generated by multiplying the two outputs of the low-pass filters. Thus,

$$\begin{aligned} e(t) &= \tfrac{1}{8}\{[A(t) + n_i(t)]^2 - n_q^2(t)\} \sin(2\Delta\phi) \\ &\quad - \tfrac{1}{4} n_q(t)[A(t) + n_i(t)] \cos(2\Delta\phi) \end{aligned} \tag{5.2–57}$$

This error signal is filtered by the loop filter, whose output is the control voltage that drives the VCO. The reader should note the similarity of the Costas loop to the PLL shown in Figure 5.2–13.

We note that the error signal into the loop filter consists of the desired term $A^2(t) \sin 2(\hat{\phi} - \phi)$ plus terms that involve signal × noise and noise × noise. These terms are similar to the two noise terms at the input to the PLL for the squaring method. In fact, if the loop filter in the Costas loop is identical to that used in the squaring loop, the two loops are equivalent. Under this condition, the probability density function of the phase error and the performance of the two loops are identical.

It is interesting to note that the optimum low-pass filter for rejecting the double-frequency terms in the Costas loop is a filter matched to the signal pulse in the information-bearing signal. If matched filters are employed for the low-pass filters, their outputs could be sampled at the bit rate at the end of each signal interval, and the discrete-time signal samples could be used to drive the loop. The use of the matched filter results in a smaller noise into the loop.

Finally, we note that, as in the squaring PLL, the output of the VCO contains a phase ambiguity of 180°, necessitating the need for differential encoding of the data prior to transmission and differential decoding at the demodulator.

Carrier estimation for multiple phase signals When the digital information is transmitted via M-phase modulation of a carrier, the methods described above can be generalized to provide the properly phased carrier for demodulation. The received M-phase signal, excluding the additive noise, may be expressed as

$$s(t) = A \cos\left[2\pi f_c t + \phi + \frac{2\pi}{M}(m-1)\right], \qquad m = 1, 2, \ldots, M \tag{5.2–58}$$

where $2\pi(m-1)/M$ represents the information-bearing component of the signal phase. The problem in carrier recovery is to remove the information-bearing component and, thus, to obtain the unmodulated carrier $\cos(2\pi f_c t + \phi)$. One method by which this can be accomplished is illustrated in Figure 5.2–16, which represents a generalization of the squaring loop. The signal is passed through an Mth-power-law device, which generates a number of harmonics of f_c. The band-pass filter selects the harmonic $\cos(2\pi M f_c t + M\phi)$ for driving the PLL. The term

$$\frac{2\pi}{M}(m-1)M = 2\pi(m-1) \equiv 0 \pmod{2\pi}, \qquad m = 1, 2, \ldots, M$$

Thus, the information is removed. The VCO output is $\sin(2\pi M f_c t + M\hat{\phi})$, so this output is divided in frequency by M to yield $\sin(2\pi f_c t + \hat{\phi})$, and phase-shifted by $\frac{1}{2}\pi$

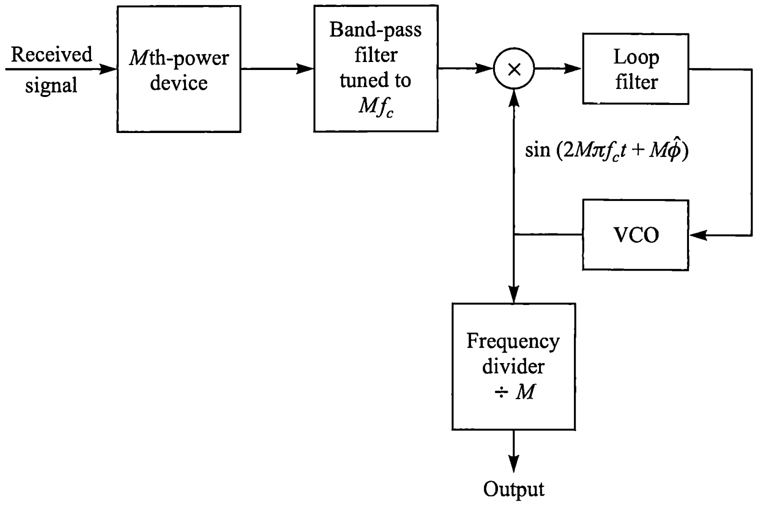


FIGURE 5.2–16
Carrier recovery with Mth-power-law device for M-ary PSK.

rad to yield $\cos(2\pi f_c t + \hat{\phi})$. These components are then fed to the demodulator. Although not explicitly shown, there is a phase ambiguity in these reference sinusoids of $360°/M$, which can be overcome by differential encoding of the data at the transmitter and differential decoding after demodulation at the receiver.

Just as in the case of the squaring PLL, the Mth-power PLL operates in the presence of noise that has been enhanced by the Mth-power-law device, which results in the output

$$y(t) = [s(t) + n(t)]^M$$

The variance of the phase error in the PLL resulting from the additive noise may be expressed in the simple form

$$\sigma_{\hat{\phi}}^2 = \frac{S_{\mathrm{ML}}^{-1}}{\gamma_L} \tag{5.2–59}$$

where γ_L is the loop SNR and S_{ML}^{-1} is the *M-phase power loss*. S_{ML} has been evaluated by Lindsey and Simon (1973) for $M = 4$ and 8.

Another method for carrier recovery in M-ary PSK is based on a generalization of the Costas loop. That method requires multiplying the received signal by M phase-shifted carriers of the form

$$\sin\left[2\pi f_c t + \hat{\phi} + \frac{\pi}{M}(k-1)\right], \qquad k = 1, 2, \ldots, M$$

low-pass–filtering each product, and then multiplying the outputs of the low-pass filters to generate the error signal. The error signal excites the loop filter, which, in turn, provides the control signal for the VCO. This method is relatively complex to implement and, consequently, has not been generally used in practice.

Comparison of decision-directed with non-decision-directed loops We note that the decision-feedback phase-locked loop (DFPLL) differs from the Costas loop only in the method by which $A(t)$ is rectified for the purpose of removing the modulation. In the Costas loop, each of the two quadrature signals used to rectify $A(t)$ is corrupted by noise. In the DFPLL, only one of the signals used to rectify $A(t)$ is corrupted by noise. On the other hand, the squaring loop is similar to the Costas loop in terms of the noise effect on the estimate $\hat{\phi}$. Consequently, the DFPLL is superior in performance to both the Costas loop and the squaring loop, provided that the demodulator is operating at error rates below 10^{-2} where an occasional decision error has a negligible effect on $\hat{\phi}$. Quantitative comparisons of the variance of the phase errors in a Costas loop to those in DFPLL have been made by Lindsey and Simon (1973), and show that the variance of the DFPLL is 4–10 times smaller for signal-to-noise ratios per bit above 0 dB.

5.3
SYMBOL TIMING ESTIMATION

In a digital communication system, the output of the demodulator must be sampled periodically at the symbol rate, at the precise sampling time instants $t_m = mT + \tau$, where T is the symbol interval and τ is a nominal time delay that accounts for the propagation time of the signal from the transmitter to the receiver. To perform this periodic sampling, we require a clock signal at the receiver. The process of extracting such a clock signal at the receiver is usually called *symbol synchronization* or *timing recovery*.

Timing recovery is one of the most critical functions that is performed at the receiver of a synchronous digital communication system. We should note that the receiver must know not only the frequency $(1/T)$ at which the outputs of the matched filters or correlators are sampled, but also where to take the samples within each symbol interval. The choice of sampling instant within the symbol interval of duration T is called the *timing phase*.

Symbol synchronization can be accomplished in one of several ways. In some communication systems, the transmitter and receiver clocks are synchronized to a master clock, which provides a very precise timing signal. In this case, the receiver must estimate and compensate for the relative time delay between the transmitted and received signals. Such may be the case for radio communication systems that operate in the very low frequency (VLF) band (below 30 kHz), where precise clock signals are transmitted from a master radio station.

Another method for achieving symbol synchronization is for the transmitter to simultaneously transmit the clock frequency $1/T$ or a multiple of $1/T$ along with the information signal. The receiver may simply employ a narrowband filter tuned to the transmitted clock frequency and, thus, extract the clock signal for sampling. This approach has the advantage of being simple to implement. There are several disadvantages, however. One is that the transmitter must allocate some of its available power to the transmission of the clock signal. Another is that some small fraction of the available channel bandwidth must be allocated for the transmission of the clock signal. In spite of these disadvantages, this method is frequently used in telephone transmission systems that employ large bandwidths to transmit the signals of many users. In such a case, the transmission of a clock signal is shared in the demodulation of the signals among the many users. Through this shared use of the clock signal, the penalty in the transmitter power and in bandwidth allocation is reduced proportionally by the number of users.

A clock signal can also be extracted from the received data signal. There are a number of different methods that can be used at the receiver to achieve self-synchronization. In this section, we treat both decision-directed and non-decision-directed methods.

5.3–1 Maximum-Likelihood Timing Estimation

Let us begin by obtaining the ML estimate of the time delay τ. If the signal is a baseband PAM waveform, it is represented as

$$r(t) = s(t; \tau) + n(t) \tag{5.3–1}$$

where

$$s(t; \tau) = \sum_n I_n g(t - nT - \tau) \tag{5.3–2}$$

As in the case of ML phase estimation, we distinguish between two types of timing estimators, decision-directed timing estimators and non-decision-directed estimators. In the former, the information symbols from the output of the demodulator are treated as the known transmitted sequence. In this case, the log-likelihood function has the form

$$\Lambda_L(\tau) = C_L \int_{T_0} r(t)s(t; \tau)\, dt \tag{5.3–3}$$

If we substitute Equation 5.3–2 into Equation 5.3–3, we obtain

$$\begin{aligned}\Lambda_L(\tau) &= C_L \sum_n I_n \int_{T_0} r(t)g(t - nT - \tau)\, dt \\ &= C_L \sum_n I_n y_n(\tau)\end{aligned} \tag{5.3–4}$$

where $y_n(t)$ is defined as

$$y_n(\tau) = \int_{T_0} r(t)g(t - nT - \tau)\, dt \tag{5.3–5}$$

A necessary condition for $\hat{\tau}$ to be the ML estimate of τ is that

$$\begin{aligned}\frac{d\Lambda_L(\tau)}{d\tau} &= \sum_n I_n \frac{d}{d\tau} \int_{T_0} r(t)g(t - nT - \tau)\, dt \\ &= \sum_n I_n \frac{d}{d\tau}[y_n(\tau)] = 0\end{aligned} \tag{5.3–6}$$

The result in Equation 5.3–6 suggests the implementation of the tracking loop shown in Figure 5.3–1. We should observe that the summation in the loop serves as the loop filter whose bandwidth is controlled by the length of the sliding window in the summation. The output of the loop filter drives the voltage-controlled clock (VCC), or voltage-controlled oscillator, which controls the sampling times for the input to the

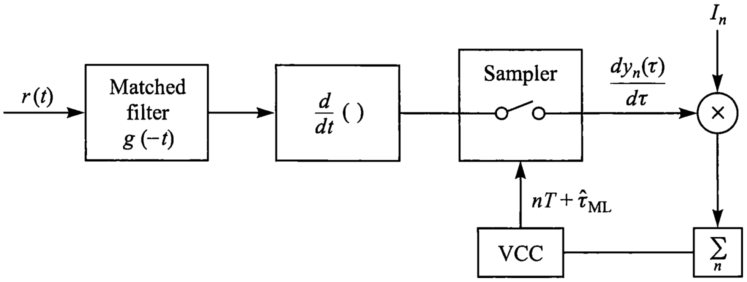


FIGURE 5.3–1
Decision-directed ML estimation of timing for baseband PAM.

loop. Since the detected information sequence $\{I_n\}$ is used in the estimation of τ, the estimate is decision-directed.

The technique described above for ML timing estimation of baseband PAM signals can be extended to carrier modulated signal formats such as QAM and PSK in a straightforward manner, by dealing with the equivalent low-pass form of the signals. Thus, the problem of ML estimation of symbol timing for carrier signals is very similar to the problem formulation for the baseband PAM signal.

5.3–2 Non-Decision-Directed Timing Estimation

A non-decision-directed timing estimate can be obtained by averaging the likelihood ratio $\Lambda(\tau)$ over the PDF of the information symbols, to obtain $\bar{\Lambda}(\tau)$, and then differentiating either $\bar{\Lambda}(\tau)$ or $\ln \bar{\Lambda}(\tau) = \bar{\Lambda}_L(\tau)$ to obtain the condition for the maximum-likelihood estimate $\hat{\tau}_{ML}$.

In the case of binary (baseband) PAM, where $I_n = \pm 1$ with equal probability, the average over the data yields

$$\bar{\Lambda}_L(\tau) = \sum_n \ln \cosh[Cy_n(\tau)] \tag{5.3–7}$$

just as in the case of the phase estimator, Since ln cosh $x \approx \frac{1}{2}x^2$ for small x, the square-law approximation

$$\bar{\Lambda}_L(\tau) \approx \tfrac{1}{2}C^2 \textstyle\sum_n y_n^2(\tau) \tag{5.3–8}$$

is appropriate for low signal-to-noise ratios. For multilevel PAM, we may approximate the statistical characteristics of the information symbols $\{I_n\}$ by the Gaussian PDF, with zero-mean and unit variance. When we average $\Lambda(\tau)$ over the Gaussian PDF, the logarithm of $\bar{\Lambda}(\tau)$ is identical to $\bar{\Lambda}_L(\tau)$ given by Equation 5.3–8. Consequently, the non-decision-directed estimate of τ may be obtained by differentiating Equation 5.3–8. The result is an approximation to the ML estimate of the delay time. The derivative of Equation 5.3–8 is

$$\frac{d}{d\tau}\sum_n y_n^2(\tau) = 2\sum_n y_n(\tau)\frac{dy_n(\tau)}{d\tau} = 0 \tag{5.3–9}$$

where $y_n(\tau)$ is given by Equation 5.3–5.

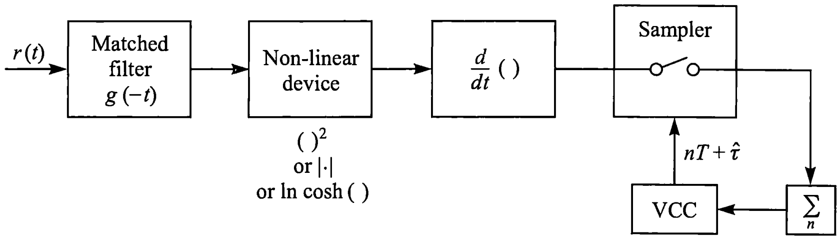


FIGURE 5.3–2
Non-decision-directed estimation of timing for binary baseband PAM.

An implementation of a tracking loop based on the derivative of $\bar{\Lambda}_L(\tau)$ given by Equation 5.3–7 is shown in Figure 5.3–2. Alternatively, an implementation of a tracking loop based on Equation 5.3–9 is illustrated in Figure 5.3–3. In both structures, we observe that the summation serves as the loop filter that drives the VCC. It is interesting to note the resemblance of the timing loop in Figure 5.3–3 to the Costas loop for phase estimation.

Early–late gate synchronizers Another non-decision-directed timing estimator exploits the symmetry properties of the signal at the output of the matched filter or correlator. To describe this method, let us consider the rectangular pulse $s(t)$, $0 \leq t \leq T$, shown in Figure 5.3–4a. The output of the filter matched to $s(t)$ attains its maximum value at time $t = T$, as shown in Figure 5.3–4b. Thus, the output of the matched filter is the time autocorrelation function of the pulse $s(t)$. Of course, this statement holds for any arbitrary pulse shape, so the approach that we describe applies in general to any signal pulse. Clearly, the proper time to sample the output of the matched filter for a maximum output is at $t = T$, i.e., at the peak of the correlation function.

In the presence of noise, the identification of the peak value of the signal is generally difficult. Instead of sampling the signal at the peak, suppose we sample early, at $t = T - \delta$ and late at $t = T + \delta$. The absolute values of the early samples $|y[m(T - \delta)]|$ and the late samples $|y[m(T + \delta)]|$ will be smaller (on the average in the presence of noise)

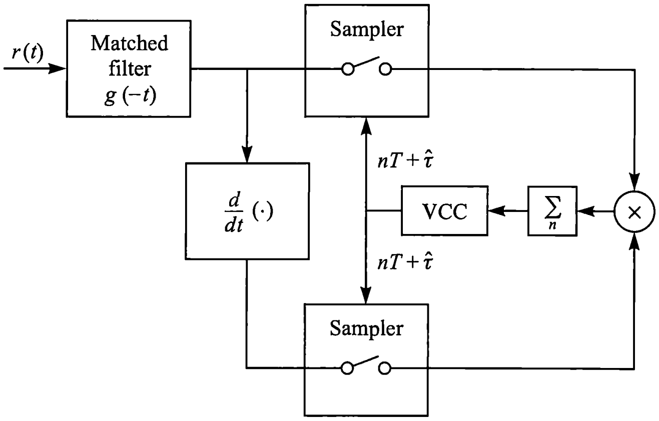


FIGURE 5.3–3
Non-decision-directed estimation of timing for baseband PAM.

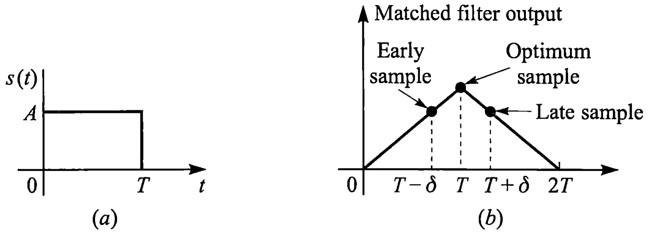


FIGURE 5.3–4
Rectangular signal pulse (a) and its matched filter output (b).

than the samples of the peak value $|y(mT)|$. Since the autocorrelation function is even with respect to the optimum sampling time $t = T$, the absolute values of the correlation function at $t = T - \delta$ and $t = T + \delta$ are equal. Under this condition, the proper sampling time is the midpoint between $t = T - \delta$ and $t = T + \delta$. This condition forms the basis for the *early–late gate symbol synchronizer.*

Figure 5.3–5 illustrates the block diagram of an early–late gate synchronizer. In this figure, correlators are used in place of the equivalent matched filters. The two correlators integrate over the symbol interval T, but one correlator starts integrating δ seconds early relative to the estimated optimum sampling time and the other integrator starts integrating δ seconds late relative to the estimated optimum sampling time. An error signal is formed by taking the difference between the absolute values of the two correlator outputs. To smooth the noise corrupting the signal samples, the error signal is passed through a low-pass filter. If the timing is off relative to the optimum sampling time, the average error signal at the output of the low-pass filter is nonzero, and the clock signal is either retarded or advanced, depending on the sign of the error. Thus, the smoothed error signal is used to drive a VCC, whose output is the desired clock signal that is used for sampling. The output of the VCC is also used as a clock signal for a symbol waveform generator that puts out the same basic pulse waveform as that of the transmitting filter. This pulse waveform is advanced and delayed and then fed to the two correlators, as shown in Figure 5.3–5. Note that if the signal pulses are rectangular, there is no need for a signal pulse generator within the tracking loop.

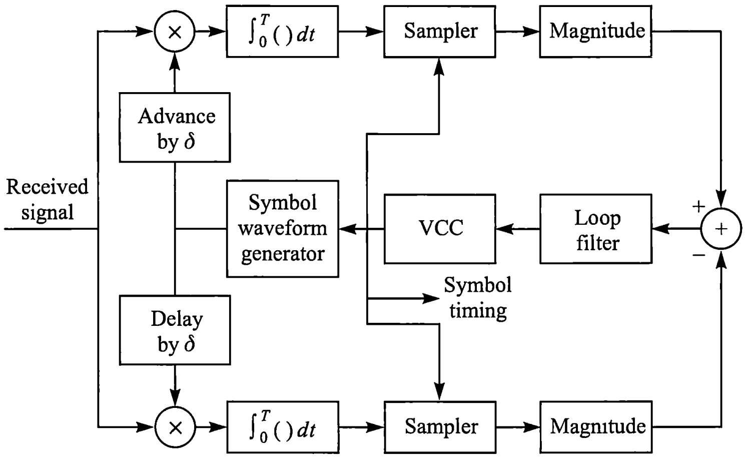


FIGURE 5.3–5
Block diagram of early–late gate synchronizer.

We observe that the early–late gate synchronizer is basically a closed-loop control system whose bandwidth is relatively narrow compared to the symbol rate $1/T$. The bandwidth of the loop determines the quality of the timing estimate. A narrowband loop provides more averaging over the additive noise and, thus, improves the quality of the estimated sampling instants, provided that the channel propagation delay is constant and the clock oscillator at the transmitter is not drifting with time (or drifting very slowly with time). On the other hand, if the channel propagation delay is changing with time and/or the transmitter clock is also drifting with time, then the bandwidth of the loop must be increased to provide for faster tracking of time variations in symbol timing.

In the tracking mode, the two correlators are affected by adjacent symbols. However, if the sequence of information symbols has zero-mean, as is the case for PAM and some other signal modulations, the contribution to the output of the correlators from adjacent symbols averages out to zero in the low-pass filter.

An equivalent realization of the early–late gate synchronizer that is somewhat easier to implement is shown in Figure 5.3–6. In this case the clock signal from the VCC is advanced and delayed by δ, and these clock signals are used to sample the outputs of the two correlators.

The early–late gate synchronizer described above is a non-decision-directed estimator of symbol timing that approximates the maximum-likelihood estimator. This assertion can be demonstrated by approximating the derivative of the log-likelihood function by the finite difference, i.e.,

$$\frac{d\bar{\Lambda}_L(\tau)}{d\tau} \approx \frac{\bar{\Lambda}_L(\tau+\delta) - \bar{\Lambda}_L(\tau-\delta)}{2\delta} \tag{5.3–10}$$

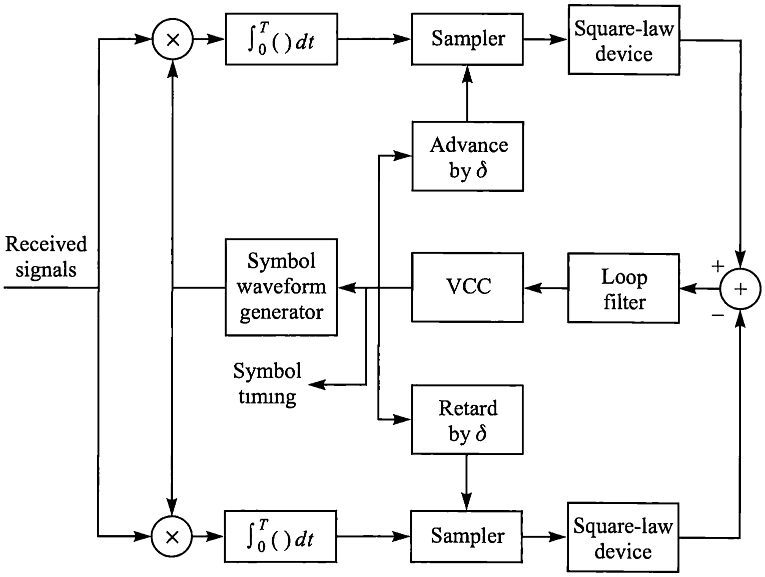


FIGURE 5.3–6
Block diagram of early–late gate synchronizer—an alternative form.

If we substitute for $\bar{\Lambda}_L(\tau)$ from Equation 5.3–8 into Equation 5.3–10, we obtain the approximation for the derivative as

$$\begin{aligned}\frac{d\bar{\Lambda}_L(\tau)}{d\tau} &= \frac{C^2}{4\delta}\sum_n \left[y_n^2(\tau+\delta) - y_n^2(\tau-\delta)\right] \\ &\approx \frac{C^2}{4\delta}\sum_n \left\{\left[\int_{T_0} r(t)g(t-nT-\tau-\delta)\,dt\right]^2\right. \\ &\quad \left. - \left[\int_{T_0} r(t)g(t-nT-\tau+\delta)\,dt\right]^2\right\}\end{aligned} \tag{5.3–11}$$

But the mathematical expression in Equation 5.3–11 basically describes the functions performed by the early–late gate symbol synchronizers illustrated in Figures 5.3–5 and 5.3–6.

5.4

JOINT ESTIMATION OF CARRIER PHASE AND SYMBOL TIMING

The estimation of the carrier phase and symbol timing may be accomplished separately as described above or jointly. Joint ML estimation of two or more signal parameters yields estimates that are as good and usually better than the estimates obtained from separate optimization of the likelihood function. In other words, the variances of the signal parameters obtained from joint optimization are less than or equal to the variance of parameter estimates obtained from separately optimizing the likelihood function.

Let us consider the joint estimation of the carrier phase and symbol timing. The log-likelihood function for these two parameters may be expressed in terms of the equivalent low-pass signals as

$$\Lambda_L(\phi,\tau) = \mathrm{Re}\left[\frac{1}{N_0}\int_{T_0} r(t)s_l^*(t;\phi,\tau)\,dt\right] \tag{5.4–1}$$

where $s_l(t;\phi,\tau)$ is the equivalent low-pass signal, which has the general form

$$s_l(t;\phi,\tau) = e^{-j\phi}\left[\sum_n I_n g(t-nT-\tau) + j\sum_n J_n w(t-nT-\tau)\right] \tag{5.4–2}$$

where $\{I_n\}$ and $\{J_n\}$ are the two information sequences.

We note that, for PAM, we may set $J_n = 0$ for all n, and the sequence $\{I_n\}$ is real. For QAM and PSK, we set $J_n = 0$ for all n and the sequence $\{I_n\}$ is complex-valued. For offset QPSK, both sequences $\{I_n\}$ and $\{J_n\}$ are nonzero and $w(t) = g(t-\frac{1}{2}T)$.

For decision-directed ML estimation of ϕ and τ, the log-likelihood function becomes

$$\Lambda_L(\phi,\tau) = \mathrm{Re}\left\{\frac{e^{j\phi}}{N_0}\sum_n \left[I_n^* y_n(\tau) - jJ_n^* x_n(\tau)\right]\right\} \tag{5.4–3}$$

where

$$\begin{aligned} y_n(\tau) &= \int_{T_0} r(t) g^*(t - nT - \tau)\, dt \\ x_n(\tau) &= \int_{T_0} r(t) w^*(t - nT - \tau)\, dt \end{aligned} \tag{5.4–4}$$

Necessary conditions for the estimates of ϕ and τ to be the ML estimates are

$$\frac{\partial \Lambda_L(\phi, \tau)}{\partial \phi} = 0, \qquad \frac{\partial \Lambda_L(\phi, \tau)}{\partial \tau} = 0 \tag{5.4–5}$$

It is convenient to define

$$A(\tau) + jB(\tau) = \frac{1}{N_0} \sum \left[I_n^* y_n(\tau) - j J_n^* x_n(\tau) \right] \tag{5.4–6}$$

With this definition, Equation 5.4–3 may be expressed in the simple form

$$\Lambda_L(\phi, \tau) = A(\tau) \cos\phi - B(\tau) \sin\phi \tag{5.4–7}$$

Now the conditions in Equation 5.4–5 for the joint ML estimates become

$$\frac{\partial \Lambda_L(\phi, \tau)}{\partial \phi} = -A(\tau) \sin\phi - B(\tau) \cos\phi = 0 \tag{5.4–8}$$

$$\frac{\partial \Lambda_L(\phi, \tau)}{\partial \tau} = \frac{\partial A(\tau)}{\partial \tau} \cos\phi - \frac{\partial B(\tau)}{\partial \tau} \sin\phi = 0 \tag{5.4–9}$$

From Equation 5.4–8, we obtain

$$\hat{\phi}_{\mathrm{ML}} = -\tan^{-1} \left[\frac{B(\hat{\tau}_{\mathrm{ML}})}{A(\hat{\tau}_{\mathrm{ML}})} \right] \tag{5.4–10}$$

The solution to Equation 5.4–9 that incorporates Equation 5.4–10 is

$$\left[A(\tau) \frac{\partial A(\tau)}{\partial \tau} + B(\tau) \frac{\partial B(\tau)}{\partial \tau} \right]_{\tau = \hat{\tau}_{\mathrm{ML}}} = 0 \tag{5.4–11}$$

The decision-directed tracking loop for QAM (or PSK) obtained from these equations is illustrated in Figure 5.4–1.

Offset QPSK requires a slightly more complex structure for joint estimation of ϕ and τ. The structure is easily derived from Equations 5.4–6 to 5.4–11.

In addition to the joint estimates given above, it is also possible to derive non-decision-directed estimates of the carrier phase and symbol timing, although we shall not pursue this approach.

We should also mention that one can combine the parameter estimation problem with the demodulation of the information sequence $\{I_n\}$. Thus, one can consider the joint maximum-likelihood estimation of $\{I_n\}$, the carrier phase ϕ, and the symbol timing parameter τ. Results on these joint estimation problems have appeared in the technical literature, e.g., Kobayashi (1971), Falconer (1976), and Falconer and Salz (1977).

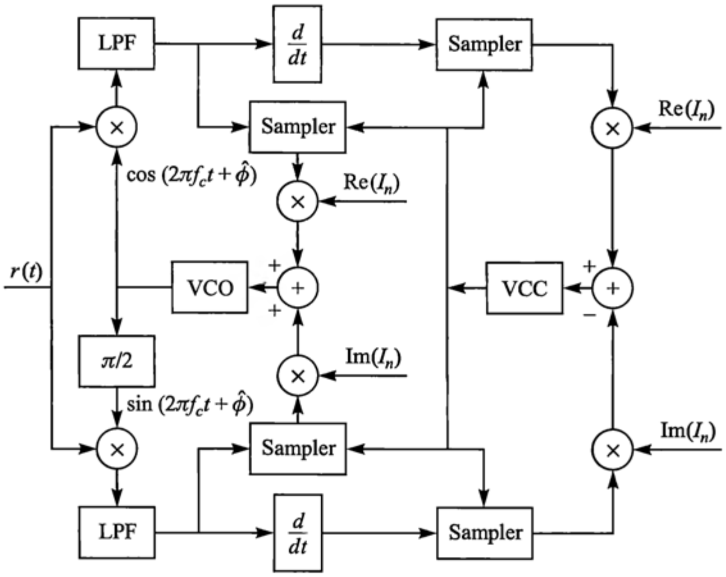


FIGURE 5.4–1
Decision-directed joint tracking loop for carrier phase and symbol timing in QAM and PSK.

5.5
PERFORMANCE CHARACTERISTICS OF ML ESTIMATORS

The quality of a signal parameter estimate is usually measured in terms of its bias and its variance. In order to define these terms, let us assume that we have a sequence of observations $(x_1\ x_2\ x_3 \cdots x_n) = \boldsymbol{x}$, with PDF $p(\boldsymbol{x}|\phi)$, from which we extract an estimate of a parameter ϕ. The bias of an estimate, say $\hat{\phi}(\boldsymbol{x})$, is defined as

$$\text{bias} = E[\hat{\phi}(\boldsymbol{x})] - \phi \tag{5.5–1}$$

where ϕ is the true value of the parameter. When $E[\hat{\phi}(\boldsymbol{x})] = \phi$, we say that the estimate is *unbiased*. The variance of the estimate $\hat{\phi}(x)$ is defined as

$$\sigma_{\hat{\phi}}^2 = E\{[\hat{\phi}(\boldsymbol{x})]^2\} - \{E[\hat{\phi}(\boldsymbol{x})]\}^2 \tag{5.5–2}$$

In general $\sigma_{\hat{\phi}}^2$ may be difficult to compute. However, a well-known result in parameter estimation (see Helstrom, 1968) is the Cramér–Rao lower bound on the mean square error defined as

$$E\{[\hat{\phi}(\boldsymbol{x}) - \phi]^2\} \geq \left\{\frac{\partial}{\partial\phi}E[\hat{\phi}(\boldsymbol{x})]\right\}^2 \bigg/ E\left\{\left[\frac{\partial}{\partial\phi}\ln p(\boldsymbol{x}|\phi)\right]^2\right\} \tag{5.5–3}$$

Note that when the estimate is unbiased, the numerator of Equation 5.5–3 is unity and the bound becomes a lower bound on the variance of $\sigma_{\hat{\phi}}^2$ of the estimate $\hat{\phi}(\boldsymbol{x})$, i.e.,

$$\sigma_{\hat{\phi}}^2 \geq 1 \bigg/ E\left\{\left[\frac{\partial}{\partial\phi}\ln p(\boldsymbol{x}|\phi)\right]^2\right\} \tag{5.5–4}$$

Since $\ln p(\boldsymbol{x}\,|\,\phi)$ differs from the log-likelihood function by a constant factor independent of ϕ, it follows that

$$\begin{aligned} E\left\{\left[\frac{\partial}{\partial\phi}\ln p(\boldsymbol{x}\,|\,\phi)\right]^2\right\} &= E\left\{\left[\frac{\partial}{\partial\phi}\ln\Lambda(\phi)\right]^2\right\} \\ &= -E\left\{\frac{\partial^2}{\partial\phi^2}\ln\Lambda(\phi)\right\} \end{aligned} \tag{5.5–5}$$

Therefore, the lower bound on the variance is

$$\sigma_{\hat{\phi}}^2 \geq 1 \Big/ E\left\{\left[\frac{\partial}{\partial\phi}\ln\Lambda(\phi)\right]^2\right\} = -1 \Big/ E\left[\frac{\partial^2}{\partial\phi^2}\ln\Lambda(\phi)\right] \tag{5.5–6}$$

This lower bound is a very useful result. It provides a benchmark for comparing the variance of any practical estimate to the lower bound. Any estimate that is unbiased and whose variance attains the lower bound is called an *efficient estimate.*

In general, efficient estimates are rare. When they exist, they are maximum-likelihood estimates. A well-known result from parameter estimation theory is that any ML parameter estimate is asymptotically (arbitrarily large number of observations) unbiased and efficient. To a large extent, these desirable properties constitute the importance of ML parameter estimates. It is also known that an ML estimate is asymptotically Gaussian distributed (with mean ϕ and variance equal to the lower bound given by Equation 5.5–6.)

In the case of the ML estimates described in this chapter for the two signal parameters, their variance is generally inversely proportional to the signal-to-noise ratio, or, equivalently, inversely proportional to the signal power multiplied by the observation interval T_0. Furthermore, the variance of the decision-directed estimates, at low error probabilities, are generally lower than the variance of non-decision-directed estimates. In fact, the performance of the ML decision-directed estimates for ϕ and τ attain the lower bound.

The following example is concerned with the evaluation of the Cramér–Rao lower bound for the ML estimate of the carrier phase.

EXAMPLE 5.5–1. The ML estimate of the phase of an unmodulated carrier was shown in Equation 5.2–11 to satisfy the condition

$$\int_{T_0} r(t)\sin(2\pi f_c t + \hat{\phi}_{\mathrm{ML}})\,dt = 0 \tag{5.5–7}$$

where

$$\begin{aligned} r(t) &= s(t;\phi) + n(t) \\ &= A\cos(2\pi f_c t + \phi) + n(t) \end{aligned} \tag{5.5–8}$$

The condition in Equation 5.5–7 was derived by maximizing the log-likelihood function

$$\Lambda_L(\phi) = \frac{2}{N_0}\int_{T_0} r(t)s(t;\phi)\,dt \tag{5.5–9}$$

The variance of $\hat{\phi}_{\text{ML}}$ is lower-bounded as

$$\begin{aligned}\sigma^2_{\hat{\phi}_{\text{ML}}} &\geq \left\{ \frac{2A}{N_0} \int_{T_0} E[r(t)] \cos(2\pi f_c t + \phi)\, dt \right\}^{-1} \\ &\geq \left\{ \frac{A^2}{N_0} \int_{T_0} dt \right\}^{-1} = \frac{N_0}{A^2 T_0} \\ &\geq \frac{N_0/T_0}{A^2} = \frac{N_0 B_{\text{eq}}}{A^2}\end{aligned} \tag{5.5–10}$$

where the factor $1/T_0$ is simply the (one-sided) equivalent noise bandwidth of the ideal integrator and $N_0 B_{\text{eq}}$ is the total noise power.

From this example, we observe that the variance of the ML phase estimate is lower-bounded as

$$\sigma^2_{\hat{\phi}_{\text{ML}}} \geq \frac{1}{\gamma_L} \tag{5.5–11}$$

where γ_L is the loop SNR. This is also the variance obtained for the phase estimate from a PLL with decision-directed estimation. As we have already observed, non-decision-directed estimates do not perform as well due to losses in the non-linearities required to remove the modulation, e.g., the squaring loss and the Mth-power loss.

Similar results can be obtained on the quality of the symbol timing estimates derived above. In addition to their dependence on the SNR, the quality of symbol timing estimates is a function of the signal pulse shape. For example, a pulse shape that is commonly used in practice is one that has a raised cosine spectrum (see Section 9.2). For such a pulse, the rms timing error ($\sigma_{\hat{\tau}}$) as a function of SNR is illustrated in Figure 5.5–1, for both decision-directed and non-decision-directed estimates. Note the significant improvement in performance of the decision-directed estimate compared with the non-decision-directed estimate. Now, if the bandwidth of the pulse is varied, the pulse shape is changed and, hence, the rms value of the timing error also changes. For example, when the bandwidth of the pulse that has a raised cosine spectrum is varied,

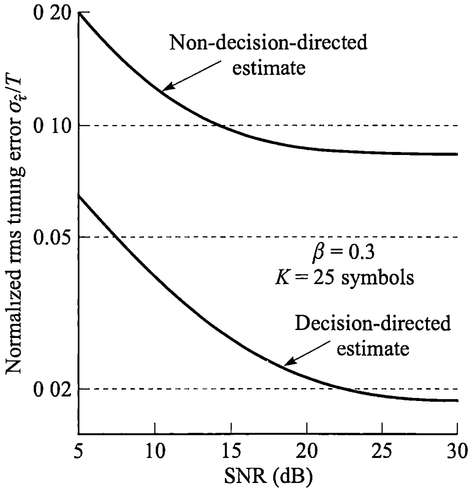


FIGURE 5.5–1
Performance of baseband symbol timing estimate for fixed signal and loop bandwidths. [*From* Synchronization Subsystems: Analysis and Design, *by L. Franks, 1981. Reprinted with permission of the author.*]

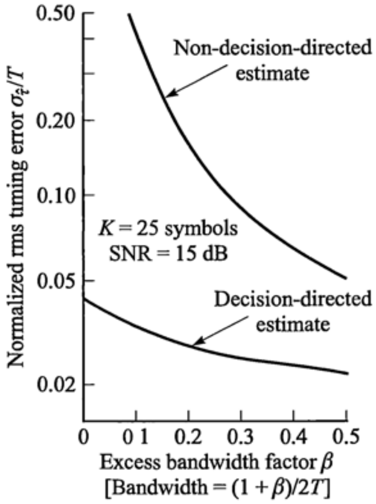


FIGURE 5.5–2
Performance of baseband symbol timing estimate for fixed SNR and fixed loop bandwidths. [*From* Synchronization Subsystems: Analysis and Design, *by L. Franks, 1981. Reprinted with permission of the author.*]

the rms timing error varies as shown in Figure 5.5–2. Note that the error decreases as the bandwidth of the pulse increases.

In conclusion, we have presented the ML method for signal parameter estimation and have applied it to the estimation of the carrier phase and symbol timing. We have also described their performance characteristics.

5.6
BIBLIOGRAPHICAL NOTES AND REFERENCES

Carrier recovery and timing synchronization are two topics that have been thoroughly investigated over the past three decades. The Costas loop was invented in 1956 and the decision-directed phase estimation methods were described in Proakis et al. (1964) and Natali and Walbesser (1969). The work on decision-directed estimation was motivated by earlier work of Price (1962a,b). Comprehensive treatments of phase-locked loops first appeared in the books by Viterbi (1966) and Gardner (1979). Books that cover carrier phase recovery and time synchronization techniques have been written by Stiffler (1971), Lindsey (1972), Lindsey and Simon (1973), Meyr and Ascheid (1990), Simon et al. (1995), Meyr et al. (1998), and Mengali and D'Andrea (1997).

A number of tutorial papers have appeared in IEEE journals on the PLL and on time synchronization. We cite, for example, the paper by Gupta (1975), which treats both analog and digital implementation of PLLs, and the paper by Lindsey and Chie (1981), which is devoted to the analysis of digital PLLs. In addition, the tutorial paper by Franks (1980) describes both carrier phase and symbol synchronization methods, including methods based on the maximum-likelihood estimation criterion. The paper by Franks is contained in a special issue of the *IEEE Transactions on Communications* (August 1980) devoted to synchronization. The paper by Mueller and Muller (1976) describes digital signal processing algorithms for extracting symbol timing and the paper by Bergmans (1995) evaluates the efficiency of data-aided timing recovery methods.

Application of the maximum-likelihood criterion to parameter estimation was first described in the context of radar parameter estimation (range and range rate).

Subsequently, this optimal criterion was applied to carrier phase and symbol timing estimation as well as to joint parameter estimation with data symbols. Papers on these topics have been published by several researchers, including Falconer (1976), Mengali (1977), Falconer and Salz (1977), and Meyers and Franks (1980).

The Cramér–Rao lower bound on the variance of a parameter estimate is derived and evaluated in a number of standard texts on detection and estimation theory, such as Helstrom (1968) and Van Trees (1968). It is also described in several books on mathematical statistics, such as the book by Cramér (1946).

PROBLEMS

5.1 Prove the relation in Equation 5.1–7.

5.2 Sketch the equivalent realization of the binary PSK receiver in Figure 5.1–1 that employs a matched filter instead of a correlator.

5.3 Suppose that the loop filter (see Equation 5.2–14) for a PLL has the transfer function

$$G(s) = \frac{1}{s + \sqrt{2}}$$

a. Determine the closed-loop transfer function $H(s)$ and indicate if the loop is stable.
b. Determine the damping factor and the natural frequency of the loop.

5.4 Consider the PLL for estimating the carrier phase of a signal in which the loop filter is specified as

$$G(s) = \frac{K}{1 + \tau_1 s}$$

a. Determine the closed-loop transfer function $H(s)$ and its gain at $f = 0$.
b. For what range of values of τ_1 and K is the loop stable?

5.5 The loop filter $G(s)$ in a PLL is implemented by the circuit shown in Figure P5.5. Determine the system function $G(s)$ and express the time constants τ_1 and τ_2 in terms of the circuit parameters.

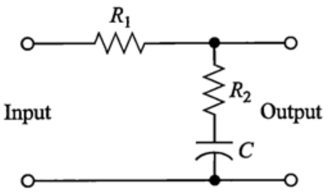


FIGURE P5.5

5.6 The loop filter $G(s)$ in a PLL is implemented with the active filter shown in Figure P5.6. Determine the system function $G(s)$ and express the time constants τ_1 and τ_2 in terms of the circuit parameters.

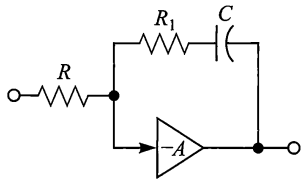


FIGURE P5.6

5.7 Show that the early–late gate synchronizer illustrated in Figure 5.3–5 is a close approximation to the timing recovery system illustrated in Figure P5.7.

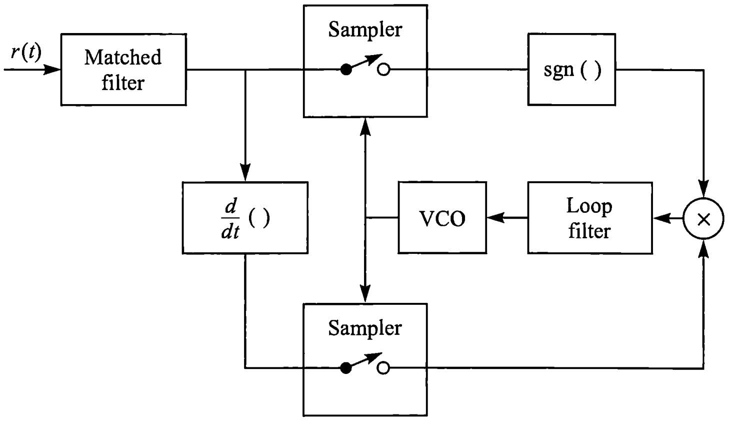


FIGURE P5.7

5.8 Based on an ML criterion, determine a carrier phase estimation method for binary on–off keying modulation.

5.9 In the transmission and reception of signals to and from moving vehicles, the transmitted signal frequency is shifted in direct proportion to the speed of the vehicle. The so-called *Doppler frequency shift* imparted to a signal that is received in a vehicle traveling at a velocity v relative to a (fixed) transmitter is given by the formula

$$f_D = \pm \frac{v}{\lambda}$$

where λ is the wavelength, and the sign depends on the direction (moving toward or moving away) that the vehicle is traveling relative to the transmitter. Suppose that a vehicle is traveling at a speed of 100 km/h relative to a base station in a mobile cellular communication system. The signal is a narrowband signal transmitted at a carrier frequency of 1 GHz.

a. Determine the Doppler frequency shift.

b. What should be the bandwidth of a Doppler frequency tracking loop if the loop is designed to track Doppler frequency shifts for vehicles traveling at speeds up to 100 km/h?

c. Suppose the transmitted signal bandwidth is 2 MHz centered at 1 GHz. Determine the Doppler frequency spread between the upper and lower frequencies in the signal.

5.10 Show that the mean value of the ML estimate in Equation 5.2–38 is ϕ, i.e., that the estimate is unbiased.

5.11 Determine the PDF of the ML phase estimate in Equation 5.2–38.

5.12 Determine the ML phase estimate for offset QPSK.

5.13 A single-sideband PAM signal may be represented as

$$u_m(t) = A_m[g_T(t)\cos 2\pi f_c t - \hat{g}_T(t)\sin 2\pi f_c t]$$

where $\hat{g}_T(t)$ is the Hilbert transform of $g_T(t)$ and A_m is the amplitude level that conveys the information. Demonstrate mathematically that a Costas loop cannot be used to demodulate the SSB PAM signal.

5.14 A carrier component is transmitted on the quadrature carrier in a communication system that transmits information via binary PSK. Hence, the received signal has the form

$$r(t) = \pm\sqrt{2P_s}\cos(2\pi f_c t + \phi) + \sqrt{2P_c}\sin(2\pi f_c t + \phi) + n(t)$$

where ϕ is the carrier phase and $n(t)$ is AWGN. The unmodulated carrier component is used as a pilot signal at the receiver to estimate the carrier phase.

a. Sketch a block diagram of the receiver, including the carrier phase estimator.

b. Illustrate mathematically the operations involved in the estimation of the carrier phase ϕ.

c. Express the probability of error for the detection of the binary PSK signal as a function of the total transmitted power $P_T = P_s + P_c$. What is the loss in performance due to the allocation of a portion of the transmitted power to the pilot signal? Evaluate the loss for $P_c/P_T = 0.1$.

5.15 Determine the signal and noise components at the input to a fourth-power ($M = 4$) PLL that is used to generate the carrier phase for demodulation of QPSK. By ignoring all noise components except those that are linear in the noise $n(t)$, determine the variance of the phase estimate at the output of the PLL.

5.16 The probability of error for binary PSK demodulation and detection when there is a carrier phase error ϕ_e is

$$P_2(\phi_e) = Q\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}\cos^2\phi_e}\right)$$

Suppose that the phase error from the PLL is modeled as a zero-mean Gaussian random variable with variance $\sigma_\phi^2 \ll \pi$. Determine the expression for the average probability of error (in integral form).

5.17 Determine the ML estimate of the time delay τ for the QAM signal of the form

$$s(t) = \mathrm{Re}[s_l(t;\tau)e^{j2\pi f_c t}]$$

where

$$s_l(t;\tau) = \sum_n I_n g(t - nT - \tau)$$

and $\{I_n\}$ is a sequence of complex-valued data.

5.18 Determine the joint ML estimate of τ and ϕ for a PAM signal.

5.19 Determine the joint ML estimate of τ and ϕ for offset QPSK.

6

An Introduction to Information Theory

This chapter deals with fundamental limits on communications. By fundamental limits we mean the study of conditions under which the two fundamental tasks in communications—compression and transmission—are possible. In this chapter we will see that for some important source and channel models, we can precisely state the limits for compression and transmission of information.

In Chapter 4, we considered the optimal detection of digitally modulated signals when transmitted through an AWGN channel. We observed that some modulation methods provide better performance than others. In particular, we observed that orthogonal signaling waveforms allow us to make the probability of error arbitrarily small by letting the number of waveforms $M \to \infty$, provided that the SNR per bit $\gamma_b > -1.6$ dB. However, if γ_b falls below -1.6 dB, then reliable communication is impossible. The value of -1.6 dB is an example of a fundamental limit for communication systems.

We begin this chapter with a study of information sources and source coding. Communication systems are designed to transmit the information generated by a source to some destination. Information sources may take a variety of different forms. For example, in radio broadcasting, the source is generally an audio source (voice or music). In TV broadcasting, the information source is a video source whose output is a moving image. The outputs of these sources are analog signals and, hence, the sources are called analog sources. In contrast, computers and storage devices, such as magnetic or optical disks, produce discrete outputs (usually binary or ASCII characters), and hence are called discrete sources.

Whether a source is analog or discrete, a digital communication system is designed to transmit information in digital form. Consequently, the output of the source must be converted to a format that can be transmitted digitally. This conversion of the source output to a digital form is generally performed by the source encoder, whose output may be assumed to be a sequence of binary digits.

In the second half of this chapter we focus on communication channels and transmission of information. We develop mathematical models for important channels and introduce two important parameters for communication channels—channel capacity and channel cutoff rate—and elaborate on their meaning and significance.

Later in Chapters 7 and 8, we consider signal waveforms generated from either binary or nonbinary sequences. We shall observe that, in general, coded waveforms offer performance advantages not only in power-limited applications where $R/W < 1$, but also in bandwidth-limited systems where $R/W > 1$.

6.1
MATHEMATICAL MODELS FOR INFORMATION SOURCES

Any information source produces an output that is random; i.e., the source output is characterized in statistical terms. Otherwise, if the source output were known exactly, there would be no need to transmit it. In this section, we consider both discrete and analog information sources, and we postulate mathematical models for each type of source.

The simplest type of a discrete source is one that emits a sequence of letters selected from a finite alphabet. For example, a binary source emits a binary sequence of the form $100101110\cdots$, where the alphabet consists of the two letters $\{0, 1\}$. More generally, a discrete information source with an alphabet of L possible letters, say $\{x_1, x_2, \ldots, x_L\}$, emits a sequence of letters selected from the alphabet.

To construct a mathematical model for a discrete source, we assume that each letter in the alphabet $\{x_1, x_2, \ldots, x_L\}$ has a given probability p_k of occurrence. That is,

$$p_k = \mathrm{P}\left[X = x_k\right], \qquad 1 \le k \le L$$

where

$$\sum_{k=1}^{L} p_k = 1$$

We consider two mathematical models of discrete sources. In the first, we assume that the output sequence from the source is statistically independent. That is, the current output letter is statistically independent of all past and future outputs. A source whose output satisfies the condition of statistical independence among output letters is said to be *memoryless*. If the source is discrete, it is called a *discrete memoryless source* (DMS). The mathematical model for a DMS is a sequence of iid random variables $\{X_i\}$.

If the output of the discrete source is statistically dependent, such as English text, we may construct a mathematical model based on statistical stationarity. By definition, a discrete source is said to be *stationary* if the joint probabilities of two sequences of length n, say, $a_1, a_2, \ldots, a_n$ and $a_{1+m}, a_{2+m}, \ldots, a_{n+m}$, are identical for all $n \ge 1$ and for all shifts m. In other words, the joint probabilities for any arbitrary length sequence of source outputs are invariant under a shift in the time origin.

An analog source has an output waveform $x(t)$ that is a sample function of a stochastic process $X(t)$. We assume that $X(t)$ is a stationary stochastic process with autocorrelation function $R_X(\tau)$ and power spectral density $\mathcal{S}_X(f)$. When $X(t)$ is a band-limited stochastic process, i.e., $\mathcal{S}_X(f) = 0$ for $|f| \ge W$, the sampling theorem may be used to represent $X(t)$ as

$$X(t) = \sum_{n=-\infty}^{\infty} X\left(\frac{n}{2W}\right) \operatorname{sinc}\left[2W\left(t - \frac{n}{2W}\right)\right] \tag{6.1–1}$$

where $\{X(n/2W)\}$ denote the samples of the process $X(t)$ taken at the sampling (Nyquist) rate of $f_s = 2W$ samples/s. Thus, by applying the sampling theorem, we may convert the output of an analog source to an equivalent discrete-time source. Then the source output is characterized statistically by the joint PDF $p(x_1, x_2, \ldots, x_m)$ for all $m \geq 1$, where $X_n = X(n/2W)$, $1 \leq n \leq m$, are the random variables corresponding to the samples of $X(t)$.

We note that the output samples $\{X(n/2W)\}$ from the stationary sources are generally continuous, and hence they cannot be represented in digital form without some loss in precision. For example, we may quantize each sample to a set of discrete values, but the quantization process results in loss of precision, and consequently the original signal cannot be reconstructed exactly from the quantized sample values. Later in this chapter, we shall consider the distortion resulting from quantization of the samples from an analog source.

6.2
A LOGARITHMIC MEASURE OF INFORMATION

To develop an appropriate measure of information, let us consider two discrete random variables X and Y with possible outcomes in the alphabets $\mathscr{X}$ and $\mathscr{Y}$, respectively. Suppose we observe some outcome $Y = y$ and we wish to determine, quantitatively, the amount of information that the occurrence of the event $Y = y$ provides about the event $X = x$. We observe that when X and Y are statistically independent, the occurrence of $Y = y$ provides no information about the occurrence of the event $X = x$. On the other hand, when X and Y are fully dependent such that the occurrence of $Y = y$ determines the occurrence of $X = x$, then the information content is simply that provided by the event $X = x$. A suitable measure that agrees with the intuitive notion of information is the logarithm of the ratio of the conditional probability

$$\mathrm{P}[X = x \mid Y = y] \triangleq \mathrm{P}[x \mid y]$$

divided by the probability

$$\mathrm{P}[X = x] \triangleq \mathrm{P}[x]$$

That is, the information content provided by the occurrence of the event $Y = y$ about the event $X = x$ is defined as

$$I(x; y) = \log \frac{\mathrm{P}[x \mid y]}{\mathrm{P}[x]} \tag{6.2–1}$$

$I(x; y)$ is called the *mutual information between x and y*. The *mutual information between random variables X and Y* is defined as the average of $I(x; y)$ and is given by

$$\begin{aligned} I(X; Y) &= \sum_{x \in \mathscr{X}} \sum_{y \in \mathscr{Y}} \mathrm{P}[X = x, Y = y]\, I(x; y) \\ &= \sum_{x \in \mathscr{X}} \sum_{y \in \mathscr{Y}} \mathrm{P}[X = x, Y = y] \log \frac{\mathrm{P}[x \mid y]}{\mathrm{P}[x]} \end{aligned} \tag{6.2–2}$$

The units of $I(X;Y)$ are determined by the base of the logarithm, which is usually selected as either 2 or e. When the base of the logarithm is 2, the units of $I(X;Y)$ are *bits*; and when the base is e, the units of $I(X;Y)$ are called *nats* (natural units). (The standard abbreviation for $\log_e$ is ln.) Since

$$\ln a = \ln 2 \log_2 a = 0.69315 \log_2 a$$

the information measured in nats is equal to ln 2 times the information measured in bits.

Some of the most important properties of the mutual information are given below. Some of these properties are proved in problems at the end of this chapter.

1. $I(X;Y) = I(Y;X)$
2. $I(X;Y) \geq 0$, with equality if and only if X and Y are independent
3. $I(X;Y) \leq \min\{|\mathscr{X}|, |\mathscr{Y}|\}$ where $|\mathscr{X}|$ and $|\mathscr{Y}|$ denote the size of the alphabets

When the random variables X and Y are statistically independent, $\mathrm{P}[x\,|y] = \mathrm{P}[x]$ and hence $I(X;Y) = 0$. On the other hand, when the occurrence of the event $Y = y$ uniquely determines the occurrence of the event $X = x$, the conditional probability in the numerator of Equation 6.2–1 is unity, hence

$$I(x;y) = \log \frac{1}{\mathrm{P}[X = x]} = -\log \mathrm{P}[X = x] \tag{6.2–3}$$

and

$$\begin{aligned} I(X;Y) &= -\sum_{x\in\mathscr{X}}\sum_{y\in\mathscr{Y}} \mathrm{P}[X = x, Y = y] \log \mathrm{P}[X = x] \\ &= -\sum_{x\in\mathscr{X}} \mathrm{P}[X = x] \log \mathrm{P}[X = x] \end{aligned} \tag{6.2–4}$$

The value of $I(X;Y)$ under this condition, which is denoted $H(X)$ and is defined by

$$H(X) = -\sum_{x\in\mathscr{X}} \mathrm{P}[X = x] \log \mathrm{P}[X = x] \tag{6.2–5}$$

is called the *entropy* of the random variable X and is a measure of uncertainty or ambiguity in X. Since knowledge of X completely removes uncertainty about it, $H(X)$ is also a measure of information that is acquired by knowledge of X, or the information content of X per source output. The unit for entropy is bits (or nats) per symbol, or per source output. Note that in the definition of entropy, we define $0 \log 0 = 0$. It is also important to note that both entropy and mutual information depend on the probabilities of the random variables and not on the values the random variables take.

If an information source is deterministic, i.e., for one value of X the probability is equal to 1 and for all other values of X the probability is equal to 0, the entropy of the source is equal to zero, i.e., there is no ambiguity in this source, and the source does not convey any information. In Problem 6.3 we show that for a DMS source with alphabet size $|\mathscr{X}|$, the entropy is maximized when all outputs are equiprobable. In this case $H(X) = \log|\mathscr{X}|$.

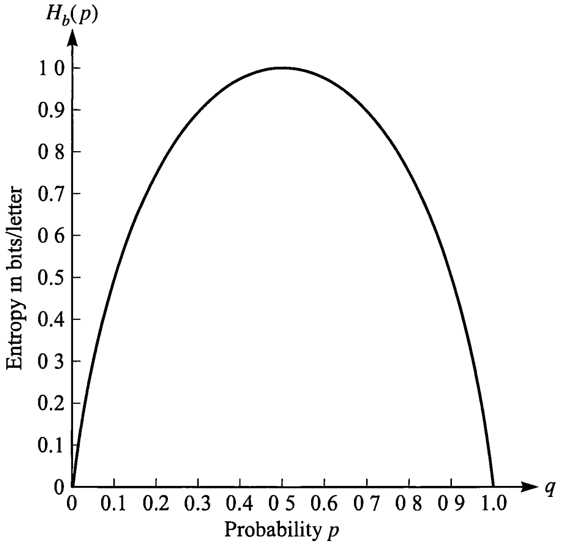


FIGURE 6.2–1
The binary entropy function.

The most important properties of the entropy functions are as follows:

1. $0 \leq H(X) \leq \log |\mathscr{X}|$
2. $I(X; X) = H(X)$
3. $I(X; Y) \leq \min\{H(X), H(Y)\}$
4. If $Y = g(X)$, then $H(Y) \leq H(X)$

EXAMPLE 6.2–1. For a binary source with probabilities p and $1 - p$ we have

$$H(X) = -p \log p - (1 - p) \log(1 - p) \tag{6.2–6}$$

This function is called the *binary entropy function* and is denoted by $H_b(p)$. A plot of $H_b(p)$ is shown in Figure 6.2–1.

Joint and Conditional Entropy

The entropy of a pair of random variables (X, Y), called the *joint entropy* of X and Y, is defined as an extension of the entropy of a single random variable as

$$H(X, Y) = -\sum_{(x,y)\in\mathscr{X}\times\mathscr{Y}} \mathrm{P}[X = x, Y = y] \log \mathrm{P}[X = x, Y = y] \tag{6.2–7}$$

When the value of random variable X is known to be x, the PMF of Y becomes $\mathrm{P}[Y = y \mid X = x]$ and the entropy of Y under this condition becomes

$$H(Y|X = x) = -\sum_{y\in\mathscr{Y}} \mathrm{P}[Y = y \mid X = x] \log \mathrm{P}[Y = y \mid X = x] \tag{6.2–8}$$

The average of this quantity over all possible values of X is denoted by $H(Y|X)$ and is called the *conditional entropy of Y given X*.

$$\begin{aligned} H(Y|X) &= \sum_{x\in\mathscr{X}} \mathrm{P}[X = x]\, H(Y|X = x) \\ &= -\sum_{(x,y)\in\mathscr{X}\times\mathscr{Y}} \mathrm{P}[X = x, Y = y] \log \mathrm{P}[Y = y \mid X = x] \end{aligned} \tag{6.2–9}$$

From Equations 6.2–7 and 6.2–9 it is easy to verify that

$$H(X, Y) = H(X) + H(Y|X) \tag{6.2–10}$$

Some of the important properties of joint and conditional entropy are summarized below.

1. $0 \leq H(X|Y) \leq H(X)$, with $H(X|Y) = H(X)$ if and only if X and Y are independent.
2. $H(X, Y) = H(X) + H(Y|X) = H(Y) + H(X|Y) \leq H(X) + H(Y)$, with equality $H(X, Y) = H(X) + H(Y)$ if and only if X and Y are independent.
3. $I(X; Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) = H(X) + H(Y) - H(X, Y)$.

The notion of joint and conditional entropy can be extended to multiple random variables. For joint entropy we have

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) = -\sum_{x_x, x_2, \ldots, x_n} & \mathrm{P}[X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n] \\ & \times \log \mathrm{P}[X_1 = x_1, X_2 = x_2, \ldots, X_n = x_{n-1}] \end{aligned} \tag{6.2–11}$$

The following relation between joint and conditional entropies is known as the *chain rule for entropies.*

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) = H(X_1) + H(X_2|X_1) + H(X_3|X_1, X_2) \\ + \cdots + H(X_n|X_1, X_2, \ldots, X_{n-1}) \end{aligned} \tag{6.2–12}$$

Using the above relation and the first property of the conditional entropy, we have

$$H(X_1, X_2, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i) \tag{6.2–13}$$

with equality if X_i's are statistically independent. If X_i's are iid, we clearly have

$$H(X_1, X_2, \ldots, X_n) = nH(X) \tag{6.2–14}$$

where $H(X)$ denotes the common value of the entropy of X_i's.

6.3
LOSSLESS CODING OF INFORMATION SOURCES

The goal of data compression is to represent a source with the fewest bits such that best recovery of the source from the compressed data is possible. Data compression can be broadly classified into *lossless* and *lossy* compression. In lossless compression the goal is to minimize the number of bits in such a way that perfect (lossless) reconstruction of the source from compressed data is possible. In lossy data compression the data are compressed subject to a maximum tolerable distortion. In this section we study the fundamental bounds for lossless compression as well as some common lossless compression algorithms.

6.3–1 The Lossless Source Coding Theorem

Let us assume that a DMS is represented by independent replicas of random variable X taking values in the set $\mathscr{X} = \{a_1, a_2, \dots, a_N\}$ with corresponding probabilities $p_1, p_2, \dots, p_N$. Let $\boldsymbol{x}$ denote an output sequence of length n for this source, where n is assumed to be large. We call this sequence a *typical sequence* if the number of occurrences of each a_i in $\boldsymbol{x}$ is roughly np_i for $1 \leq i \leq N$. The set of typical sequences is denoted by $\mathcal{A}$.

The law of large numbers, reviewed in Section 2.5, states that with high probability approaching 1 as $n \to \infty$, outputs of any DMS will be typical. Since the number of occurrences of a_i in $\boldsymbol{x}$ is roughly np_i and the source is memoryless, we have

$$\begin{aligned} \log \mathrm{P}[\boldsymbol{X} = \boldsymbol{x}] &\approx \log \prod_{i=1}^{N} (p_i)^{np_i} \\ &= \sum_{i=1}^{N} np_i \log p_i \\ &= -nH(X) \end{aligned} \tag{6.3–1}$$

Hence,

$$\mathrm{P}[\boldsymbol{X} = \boldsymbol{x}] \approx 2^{-nH(X)} \tag{6.3–2}$$

This states that all typical sequences have roughly the same probability, and this common probability is $2^{-nH(X)}$.

Since the probability of the typical sequences, for large n, is very close to 1, we conclude that the number of typical sequences, i.e., the cardinality of $\mathcal{A}$, is roughly

$$|\mathcal{A}| \approx 2^{nH(X)} \tag{6.3–3}$$

This discussion shows that for large n, a subset of all possible sequences, called the typical sequences, is almost certain to occur. Therefore, for transmission of source outputs it is sufficient to consider only this subset. Since the number of typical sequences is $2^{nH(X)}$, for their transmission $nH(X)$ bits are sufficient, and therefore the number of required bits per source output, i.e., the transmission rate, is given by

$$R \approx \frac{nH(X)}{n} = H(X) \qquad \text{bits per transmission} \tag{6.3–4}$$

The informal argument given above can be made rigorous (see the books by Cover and Thomas (2006) and Gallager (1968)) in the following theorem first stated by Shannon (1948).

> **SHANNON'S FIRST THEOREM (LOSSLESS SOURCE CODING THEOREM)** Let X denote a DMS with entropy X. There exists a lossless source code for this source at any rate R if $R > H(X)$. There exists no lossless code for this source at rates less than $H(X)$.

This theorem sets a fundamental limit on lossless source coding and shows that the entropy of a DMS, which was defined previously based on intuitive reasoning, plays a fundamental role in lossless compression of information sources.

Discrete Stationary Sources

We have seen that the entropy of a DMS sets a fundamental limit on the rate at which the source can be losslessly compressed. In this section, we consider discrete sources for which the sequence of output letters is statistically dependent. We limit our treatment to sources that are statistically stationary.

Let us evaluate the entropy of any sequence of letters from a stationary source. From the chain rule for the entropies stated in Equation 6.2–12, the entropy of a block of random variables $X_1 X_2 \cdots X_k$ is

$$H(X_1 X_2 \cdots X_k) = \sum_{i=1}^{k} H(X_i | X_1 X_2 \cdots X_{i-1}) \tag{6.3–5}$$

where $H(X_i | X_1 X_2 \cdots X_{i-1})$ is the conditional entropy of the ith symbol from the source, given the previous $i - 1$ symbols. The entropy per letter for the k-symbol block is defined as

$$H_k(X) = \frac{1}{k} H(X_1 X_2 \cdots X_k) \tag{6.3–6}$$

We define the *entropy rate* of a stationary source as the entropy per letter in Equation 6.3–6 in the limit as $k \to \infty$. That is,

$$H_\infty(X) \triangleq \lim_{k\to\infty} H_k(X) = \lim_{k\to\infty} \frac{1}{k} H(X_1 X_2 \cdots X_k) \tag{6.3–7}$$

The existence of this limit is established below.

As an alternative, we may define the entropy rate of the source in terms of the conditional entropy $H(X_k | X_1 X_2 \cdots X_{k-1})$ in the limit as k approaches infinity. Fortunately, this limit also exists and is identical to the limit in Equation 6.3–7. That is,

$$H_\infty(X) = \lim_{k\to\infty} H(X_k | X_1 X_2 \cdots X_{k-1}) \tag{6.3–8}$$

This result is also established below. Our development follows the approach in Gallager (1968).

First, we show that

$$H(X_k | X_1 X_2 \cdots X_{k-1}) \le H(X_{k-1} | X_1 X_2 \cdots X_{k-2}) \tag{6.3–9}$$

for $k \ge 2$. From our previous result that conditioning on a random variable cannot increase entropy, we have

$$H(X_k | X_1 X_2 \cdots X_{k-1}) \le H(X_k | X_2 X_3 \cdots X_{k-1}) \tag{6.3–10}$$

From the stationarity of the source, we have

$$H(X_k | X_2 X_3 \cdots X_{k-1}) = H(X_{k-1} | X_1 X_2 \cdots X_{k-2}) \tag{6.3–11}$$

Hence, Equation 6.3–9 follows immediately. This result demonstrates that $H(X_k | X_1 X_2 \cdots X_{k-1})$ is a nonincreasing sequence in k.

Second, we have the result

$$H_k(X) \ge H(X_k | X_1 X_2 \cdots X_{k-1}) \tag{6.3–12}$$

which follows immediately from Equations 6.3–5 and 6.3–6 and the fact that the last term in the sum of Equation 6.3–5 is a lower bound on each of the other $k-1$ terms.

Third, from the definition of $H_k(X)$, we may write

$$\begin{aligned} H_k(X) &= \frac{1}{k}[H(X_1X_2\cdots X_{k-1}) + H(X_k|X_1\cdots X_{k-1})] \\ &= \frac{1}{k}\,[(k-1)H_{k-1}(X) + H(X_k|X_1\cdots X_{k-1})] \\ &\leq \frac{k-1}{k}H_{k-1}(X) + \frac{1}{k}H_k(X) \end{aligned} \tag{6.3–13}$$

which reduces to

$$H_k(X) \leq H_{k-1}(X) \tag{6.3–14}$$

Hence, $H_k(X)$ is a nonincreasing sequence in k.

Since $H_k(X)$ and the conditional entropy $H(X_k|X_1\cdots X_{k-1})$ are both nonnegative and nonincreasing with k, both limits must exist. Their limiting forms can be established by using Equations 6.3–5 and 6.3–6 to express $H_{k+j}(X)$ as

$$\begin{aligned} H_{k+j}(X) &= \frac{1}{k+j}\;H(X_1X_2\cdots X_{k-1}) \\ &+ \frac{1}{k+j}\;\big[H(X_k|X_1\cdots X_{k-1}) + H(X_{k+1}|X_1\cdots X_k) \\ &+ \cdots + H(X_{k+j}|X_1\cdots X_{k+j-1})\big] \end{aligned} \tag{6.3–15}$$

Since the conditional entropy is nonincreasing, the first term in the square brackets serves as an upper bound on the other terms. Hence,

$$H_{k+j}(X) \leq \frac{1}{k+j}\;H(X_1X_2\cdots X_{k-1}) + \frac{j+1}{k+j}\;H(X_k|X_1X_2\cdots X_{k-1}) \tag{6.3–16}$$

For a fixed k, the limit of Equation 6.3–16 as $j \to \infty$ yields

$$H_\infty(X) \leq H(X_k|X_1X_2\cdots X_{k-1}) \tag{6.3–17}$$

But Equation 6.3–17 is valid for all k; hence, it is valid for $k \to \infty$. Therefore,

$$H_\infty(X) \leq \lim_{k\to\infty} H(X_k|X_1X_2\cdots X_{k-1}) \tag{6.3–18}$$

On the other hand, from Equation 6.3–12, we obtain in the limit as $k \to \infty$

$$H_\infty(X) \geq \lim_{k\to\infty} H(X_k|X_1X_2\cdots X_{k-1}) \tag{6.3–19}$$

which establishes Equation 6.3–8.

From the discussion above the entropy rate of a discrete stationary source is defined as

$$H_\infty(X) = \lim_{k\to\infty} H(X_k|X_1, X_2, \ldots, X_{k-1}) = \lim_{k\to\infty} \frac{1}{k} H(X_1, X_2, \ldots, X_k) \quad (6.3\text{–}20)$$

It is clear from above that if the source is memoryless, the entropy rate is equal to the entropy of the source.

For discrete stationary sources, the entropy rate is the fundamental rate for compression of the source such that lossless recovery is possible. Therefore, a lossless coding theorem for discrete stationary sources, similar to the one for discrete memoryless sources, exists that states lossless compression of the source at rates above the entropy rate is possible, but lossless compression at rates below the entropy rate is impossible.

6.3–2 Lossless Coding Algorithms

In this section we study two main approaches for lossless compression of discrete information sources—the Huffman coding algorithm and the Lempel-Ziv algorithm. The Huffman coding algorithm is an example of a *variable-length coding algorithm*, and the Lempel-Ziv algorithm is a *fixed-length coding algorithm.*

Variable-Length Source Coding

When the source symbols are not equally probable, an efficient encoding method is to use variable-length code words. An example of such encoding is the Morse code, which dates back to the nineteenth century. In the Morse code, the letters that occur more frequently are assigned short code words, and those that occur infrequently are assigned long code words. Following this general philosophy, we may use the probabilities of occurrence of the different source letters in the selection of the code words. The problem is to devise a method for selecting and assigning the code words to source letters. This type of encoding is called *entropy coding.*

For example, suppose that a DMS with output letters a_1, a_2, a_3, a_4 and corresponding probabilities $P(a_1) = \frac{1}{2}$, $P(a_2) = \frac{1}{4}$, and $P(a_3) = P(a_4) = \frac{1}{8}$ is encoded as shown in Table 6.3–1. Code I is a variable-length code that has a basic flaw. To see the flaw, suppose we are presented with the sequence $001001\cdots$. Clearly, the first symbol corresponding to 00 is a_2. However, the next 4 bits are ambiguous (not *uniquely decodable).* They may be decoded either as a_4a_3 or as $a_1a_2a_1$. Perhaps, the ambiguity can be

TABLE 6.3–1
Variable-Length Codes.

Letter	P$[a_k]$	Code I	Code II	Code III
a_1	$\frac{1}{2}$	1	0	0
a_2	$\frac{1}{4}$	00	10	01
a_3	$\frac{1}{8}$	01	110	011
a_4	$\frac{1}{8}$	10	111	111

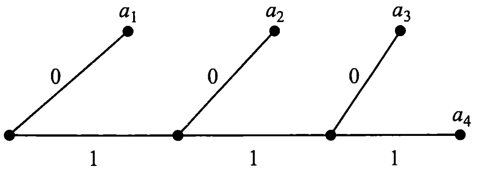


FIGURE 6.3–1
Code tree for code II in Table 6.3–1.

resolved by waiting for additional bits, but such a decoding delay is highly undesirable. We shall consider only codes that are decodable instantaneously, i.e., without any decoding delay. Such codes are called *instantaneous codes*.

Code II in Table 6.3–1 is uniquely decodable and instantaneous. It is convenient to represent the code words in this code graphically as terminal nodes of a tree, as shown in Figure 6.3–1. We observe that the digit 0 indicates the end of a code word for the first three code words. This characteristic plus the fact that no code word is longer than three binary digits makes this code instantaneously decodable. Note that no code word in this code is a prefix of any other code word. In general, the *prefix condition* requires that for a given code word $\boldsymbol{c}_k$ of length k having elements $(b_1, b_2, \ldots, b_k)$, there is no other code word of length $l < k$ with elements $(b_1, b_2, \ldots, b_l)$ for $1 \le l \le k-1$. In other words, there is no code word of length $l < k$ that is identical to the first l binary digits of another code word of length $k > l$. This property makes the code words uniquely and instantaneously decodable.

Code III given in Table 6.3–1 has the tree structures shown in Figure 6.3–2. We note that in this case the code is uniquely decodable but not instantaneously decodable. Clearly, this code does not satisfy the prefix condition.

Our main objective is to devise a systematic procedure for constructing uniquely decodable variable-length codes that are efficient in the sense that the average number of bits per source letter, defined as the quantity

$$\bar{R} = \sum_{k=1}^{L} n_k P(a_k) \tag{6.3–21}$$

is minimized. The conditions for the existence of a code that satisfies the prefix condition are given by the Kraft inequality.

The Kraft Inequality

The Kraft inequality states that a necessary and sufficient condition for the existence of a binary code with code words having lengths $n_1 \le n_2 \le \cdots \le n_L$ that satisfy the prefix condition is

$$\sum_{k=1}^{L} 2^{-n_k} \le 1 \tag{6.3–22}$$

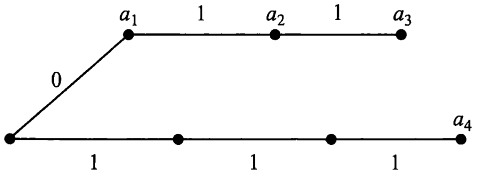


FIGURE 6.3–2
Code tree for code III in Table 6.3–1.

First, we prove that Equation 6.3–22 is a sufficient condition for the existence of a code that satisfies the prefix condition. To construct such a code, we begin with a full binary tree of order $n = n_L$ that has 2^n terminal nodes and two nodes of order k stemming from each node of order $k - 1$, for each k, $1 \le k \le n$. Let us select any node of order n_1 as the first code word $\boldsymbol{c}_1$. This choice eliminates 2^{n-n_1} terminal nodes (or the fraction 2^{-n_1} of the 2^n terminal nodes). From the remaining available nodes of order n_2, we select one node for the second code word $\boldsymbol{c}_2$. This choice eliminates 2^{n-n_2} terminal nodes (or the fraction 2^{-n_2} of the 2^n terminal nodes). This process continues until the last code word is assigned at terminal node $n = n_L$. Since, at the node of order $j < L$, the fraction of the number of terminal nodes eliminated is

$$\sum_{k=1}^{j} 2^{-n_k} < \sum_{k=1}^{L} 2^{-n_k} \le 1 \tag{6.3–23}$$

there is always a node of order $k > j$ available to be assigned to the next code word. Thus, we have constructed a code tree that is embedded in the full tree of 2^n nodes as illustrated in Figure 6.3–3, for a tree having 16 terminal nodes and a source output consisting of five letters with $n_1 = 1, n_2 = 2, n_3 = 3$, and $n_4 = n_5 = 4$.

To prove that Equation 6.3–22 is a necessary condition, we observe that in the code tree of order $n = n_L$, the number of terminal nodes eliminated from the total number of 2^n terminal nodes is

$$\sum_{k=1}^{L} 2^{n-n_k} \le 2^n \tag{6.3–24}$$

Hence,

$$\sum_{k=1}^{L} 2^{-n_k} \le 1 \tag{6.3–25}$$

and the proof of Kraft inequality is complete.

The Kraft inequality may be used to prove the following version of the lossless source coding theorem, which applies to codes that satisfy the prefix condition.

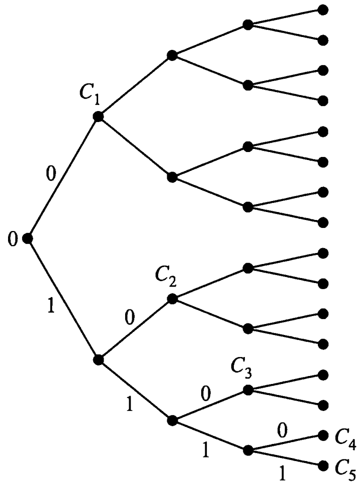


FIGURE 6.3–3
Construction of binary tree code embedded in a full tree.

SOURCE CODING THEOREM FOR PREFIX CODES Let X be a DMS with finite entropy $H(X)$ and output letters a_i, $1 \leq i \leq N$, with corresponding probabilities of occurrence p_i, $1 \leq i \leq N$. It is possible to construct a code that satisfies the prefix condition and has an average length $\bar{R}$ that satisfies the inequalities

$$H(X) \leq \bar{R} < H(X) + 1 \tag{6.3–26}$$

To establish the lower bound in Equation 6.3–26, we note that for code words that have length n_i, $1 \leq i \leq N$, the difference $H(X) - \bar{R}$ may be expressed as

$$\begin{aligned} H(X) - \bar{R} &= \sum_{i=1}^{N} p_i \log_2 \frac{1}{p_i} - \sum_{i=1}^{N} p_i n_i \\ &= \sum_{i=1}^{N} p_i \log_2 \frac{2^{-n_i}}{p_i} \end{aligned} \tag{6.3–27}$$

Use of the inequality $\ln x \leq x - 1$ in Equation 6.3–27 yields

$$\begin{aligned} H(X) - \bar{R} &\leq (\log_2 e) \sum_{i=1}^{N} p_i \left(\frac{2^{-n_i}}{p_i} - 1 \right) \\ &\leq (\log_2 e) \left(\sum_{i=1}^{N} 2^{-n_i} - 1 \right) \leq 0 \end{aligned} \tag{6.3–28}$$

where the last inequality follows from the Kraft inequality. Equality holds if and only if $p_i = 2^{-n_i}$ for $1 \leq i \leq N$.

The upper bound in Equation 6.3–26 may be established under the constraint that n_i, $1 \leq i \leq N$, are integers, by selecting the $\{n_i\}$ such that $2^{-n_i} \leq p_i < 2^{-n_i+1}$. But if the terms $p_i \geq 2^{-n_i}$ are summed over $1 \leq i \leq N$, we obtain the Kraft inequality, for which we have demonstrated that there exists a code that satisfies the prefix condition. On the other hand, if we take the logarithm of $p_i < 2^{-n_i+1}$, we obtain

$$\log p_i < -n_i + 1 \tag{6.3–29}$$

or, equivalently,

$$n_i < 1 - \log p_i \tag{6.3–30}$$

If we multiply both sides of Equation 6.3–30 by p_i and sum over $1 \leq i \leq N$, we obtain the desired upper bound given in Equation 6.3–26. This completes the proof of Equation 6.3–26.

We have now established that variable-length codes that satisfy the prefix condition are efficient source codes for any DMS with source symbols that are not equally probable. Let us now describe an algorithm for constructing such codes.

The Huffman Coding Algorithm

Huffman (1952) devised a variable-length encoding algorithm, based on the source letter probabilities $P(x_i)$, $i = 1, 2, \ldots, L$. This algorithm is optimum in the sense that the average number of binary digits required to represent the source symbols is a minimum, subject to the constraint that the code words satisfy the prefix condition, as defined above, which allows the received sequence to be uniquely and instantaneously decodable. We illustrate this encoding algorithm by means of two examples.

EXAMPLE 6.3-1. Consider a DMS with seven possible symbols $x_1, x_2, \ldots, x_7$ having the probabilities of occurrence illustrated in Figure 6.3–4. We have ordered the source symbols in decreasing order of the probabilities, i.e., $P(x_1) > P(x_2) > \cdots > P(x_7)$. We begin the encoding process with the two least probable symbols x_6 and x_7. These two symbols are tied together as shown in Figure 6.3–4, with the upper branch assigned a 0 and the lower branch assigned a 1. The probabilities of these two branches are added together at the node where the two branches meet to yield the probability 0.01. Now we have the source symbols $x_1, \ldots, x_5$ plus a new symbol, say x_6', obtained by combining x_6 and x_7. The next step is to join the two least probable symbols from the set $x_1, x_2, x_3, x_4, x_5, x_6'$. These are x_5 and x_6', which have a combined probability of 0.05. The branch from x_5 is assigned a 0 and the branch from x_6' is assigned a 1. This procedure continues until we exhaust the set of possible source letters. The result is a code tree with branches that contain the desired code words. The code words are obtained by beginning at the rightmost node in the tree and proceeding to the left. The resulting code words are listed in Figure 6.3–4. The average number of binary digits per symbol for this code is $\bar{R} = 2.21$ bits per symbol. The entropy of the source is 2.11 bits per symbol.

We make the observation that the code is not necessarily unique. For example, at the next to the last step in the encoding procedure, we have a tie between x_1 and x_3', since these symbols are equally probable. At this point, we chose to pair x_1 with x_2. An alternative is to pair x_2 with x_3'. If we choose this pairing, the resulting code is illustrated in Figure 6.3–5. The average number of bits per source symbol for this code is also 2.21. Hence, the resulting codes are equally efficient. Secondly, the assignment of a 0 to the upper branch and a 1 to the lower (less probable) branch is arbitrary. We may

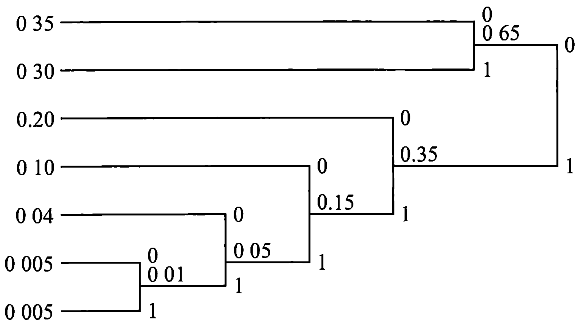


FIGURE 6.3–4
An example of variable-length source encoding for a DMS.

Letter	Probability	Self-information	Code
x_1	0.35	1.5146	00
x_2	0.30	1.7370	01
x_3	0.20	2.3219	10
x_4	0.10	3.3219	110
x_5	0.04	4.6439	1110
x_6	0.005	7.6439	11110
x_7	0.005	7.6439	11111
	$H(X) = 2.11$	$\bar{R} = 2.21$	

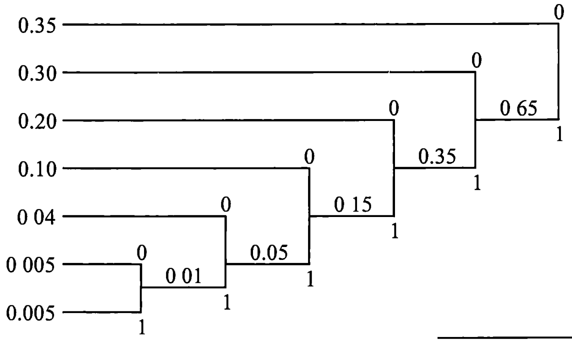

0.35
0.30
0.20
0.10
0.04
0.005
0.005
0
0
0
0
0
0
1
1
1
1
1
1
0.65
0.35
0.15
0.05
0.01

FIGURE 6.3–5
An alternative code for the DMS in Example 6.3–1.

Letter	Code
x_1	0
x_2	10
x_3	110
x_4	1110
x_5	11110
x_6	111110
x_7	111111

$\bar{R} = 2.21$

simply reverse the assignment of a 0 and 1 and still obtain an efficient code satisfying the prefix condition.

EXAMPLE 6.3–2. As a second example, let us determine the Huffman code for the output of a DMS illustrated in Figure 6.3–6. The entropy of this source is $H(X) = 2.63$ bits per symbol. The Huffman code as illustrated in Figure 6.3–6 has an average length of $\bar{R} = 2.70$ bits per symbol. Hence, its efficiency is 0.97.

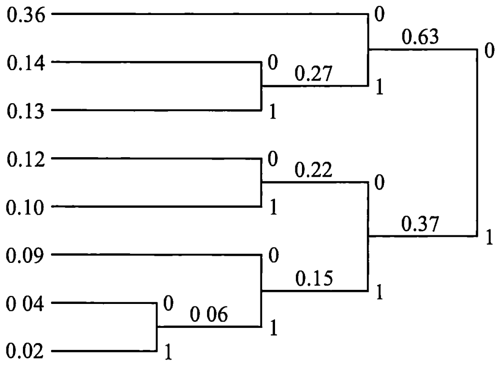

0.36
0.14
0.13
0.12
0.10
0.09
0.04
0.02
0
1
0
1
0
1
0
1
0
1
0
1
0
1
0.63
0.27
0.22
0.37
0.15
0.06

FIGURE 6.3–6
Huffman code for Example 6.3–2.

Letter	Code
x_1	00
x_2	010
x_3	011
x_4	100
x_5	101
x_6	110
x_7	1110
x_8	1111

$H(X) = 2.63$ $\bar{R} = 2.70$

TABLE 6.3–2
Huffman code for Example 6.3–3

Letter	Probability	Self-information	Code
x_1	0.45	1.156	1
x_2	0.35	1.520	00
x_3	0.20	2.330	01
	$H(X) = 1.513$ bits/letter		
	$\bar{R}_1 = 1.55$ bits/letter		
	Efficiency = 97.6%		

The variable-length encoding (Huffman) algorithm described in the above examples generates a prefix code having an $\bar{R}$ that satisfies Equation 6.3–26. However, instead of encoding on a symbol-by-symbol basis, a more efficient procedure is to encode blocks of J symbols at a time. In such a case, the bounds in Equation 6.3–26 become

$$JH(X) \leq \bar{R}_J < JH(X) + 1, \tag{6.3–31}$$

since the entropy of a J-symbol block from a DMS is $JH(X)$, and $\bar{R}_J$ is the average number of bits per J-symbol blocks. If we divide Equation 6.3–31 by J, we obtain

$$H(X) \leq \frac{\bar{R}_J}{J} < H(X) + \frac{1}{J} \tag{6.3–32}$$

where $\bar{R}_J/J \equiv \bar{R}$ is the average number of bits per source symbol. Hence $\bar{R}$ can be made as close to $H(X)$ as desired by selecting J sufficiently large.

EXAMPLE 6.3–3. The output of a DMS consists of *letters* x_1, x_2, and x_3 with probabilities 0.45, 0.35, and 0.20, respectively. The entropy of this source is $H(X) = 1.513$ bits per symbol. The Huffman code for this source, given in Table 6.3–2, requires $\bar{R}_1 = 1.55$ bits per symbol and results in an efficiency of 97.6 percent. If pairs of symbols are encoded by means of the Huffman algorithm, the resulting code is as given in Table 6.3–3. The entropy of the source output for pairs of letters is $2H(X) = 3.026$ bits per symbol

TABLE 6.3–3
Huffman code for encoding pairs of letters

Letter pair	Probability	Self-information	Code
x_1x_1	0.2025	2.312	10
x_1x_2	0.1575	2.676	001
x_2x_1	0.1575	2.676	010
x_2x_2	0.1225	3.039	011
x_1x_3	0.09	3.486	111
x_3x_1	0.09	3.486	0000
x_2x_3	0.07	3.850	0001
x_3x_2	0.07	3.850	1100
x_3x_3	0.04	4.660	1101
	$2H(X) = 3.026$ bits/letter pair		
	$\bar{R}_2 = 3.0675$ bits/letter pair		
	$\frac{1}{2}\bar{R}_2 = 1.534$ bits/letter		
	Efficiency = 98.6%		

pair. On the other hand, the Huffman code requires $\bar{R}_2 = 3.0675$ bits per symbol pair. Thus, the efficiency of the encoding increases to $2H(X)/\bar{R}_2 = 0.986$ or, equivalently, to 98.6 percent.

In summary, we have demonstrated that efficient encoding for a DMS may be done on a symbol-by-symbol basis using a variable-length code based on the Huffman algorithm. Furthermore, the efficiency of the encoding procedure is increased by encoding blocks of J symbols at a time. Thus, the output of a DMS with entropy $H(X)$ may be encoded by a variable-length code with an average number of bits per source letter that approaches $H(X)$ as closely as desired.

The Huffman coding algorithm can be applied to discrete stationary sources as well as discrete memoryless sources. Suppose we have a discrete stationary source that emits J letters with $H_J(X)$ as the entropy per letter. We can encode the sequence of J letters with a variable-length Huffman code that satisfies the prefix condition by following the procedure described above. The resulting code has an average number of bits for the J-letter block that satisfies the condition

$$H(X_1 \cdots X_J) \le \bar{R}_J < H(X_1 \cdots X_J) + 1 \tag{6.3–33}$$

By dividing each term of Equation 6.3–33 by J, we obtain the bounds on the average number $\bar{R} = \bar{R}_J/J$ of bits per source letter as

$$H_J(X) \le \bar{R} < H_J(X) + \frac{1}{J} \tag{6.3–34}$$

By increasing the block size J, we can approach $H_J(X)$ arbitrarily closely, and in the limit as $J \to \infty$, $\bar{R}$ satisfies

$$H_\infty(X) \le \bar{R} < H_\infty(X) + \epsilon \tag{6.3–35}$$

where ϵ approaches zero as $1/J$. Thus, efficient encoding of stationary sources is accomplished by encoding large blocks of symbols into code words. We should emphasize, however, that the design of the Huffman code requires knowledge of the joint PDF for the J-symbol blocks.

The Lempel–Ziv Algorithm

From our preceding discussion, we have observed that the Huffman coding algorithm yields optimal source codes in the sense that the code words satisfy the prefix condition and the average block length is a minimum. To design a Huffman code for a DMS, we need to know the probabilities of occurrence of all the source letters. In the case of a discrete source with memory, we must know the joint probabilities of blocks of length $n \ge 2$. However, in practice, the statistics of a source output are often unknown. In principle, it is possible to estimate the probabilities of the discrete source output by simply observing a long information sequence emitted by the source and obtaining the probabilities empirically. Except for the estimation of the marginal probabilities $\{p_k\}$, corresponding to the frequency of occurrence of the individual source output letters, the computational complexity involved in estimating joint probabilities is extremely high. Consequently, the application of the Huffman coding method to source coding for many real sources with memory is generally impractical.

In contrast to the Huffman coding algorithm, the Lempel–Ziv source coding algorithm does not require the source statistics. Hence, the Lempel–Ziv algorithm belongs to the class of *universal source coding algorithms*. It is a variable-to-fixed-length algorithm, where the encoding is performed as described below.

In the Lempel–Ziv algorithm, the sequence at the output of the discrete source is parsed into variable-length blocks, which are called *phrases*. A new phrase is introduced every time a block of letters from the source differs from some previous phrase in the last letter. The phrases are listed in a dictionary, which stores the location of the existing phrases. In encoding a new phrase, we simply specify the location of the existing phrase in the dictionary and append the new letter.

As an example, consider the binary sequence

$$10101101001001110101000011001110101100011011$$

Parsing the sequence as described above produces the following phrases:

$$1, 0, 10, 11, 01, 00, 100, 111, 010, 1000, 011, 001, 110, 101, 10001, 1011$$

We observe that each phrase in the sequence is a concatenation of a previous phrase with a new output letter from the source. To encode the phrases, we construct a dictionary as shown in Table 6.3–4. The dictionary locations are numbered consecutively, beginning with 1 and counting up, in this case to 16, which is the number of phrases in the sequence. The different phrases corresponding to each location are also listed, as shown. The code words are determined by listing the dictionary location (in binary form) of the previous phrase that matches the new phrase in all but the last location. Then, the new output letter is appended to the dictionary location of the previous phrase. Initially, the location 0000 is used to encode a phrase that has not appeared previously.

TABLE 6.3–4
Dictionary for Lempel-Ziv algorithm

	Dictionary location	Dictionary contents	Code word
1	0001	1	00001
2	0010	0	00000
3	0011	10	00010
4	0100	11	00011
5	0101	01	00101
6	0110	00	00100
7	0111	100	00110
8	1000	111	01001
9	1001	010	01010
10	1010	1000	01110
11	1011	011	01011
12	1100	001	01101
13	1101	110	01000
14	1110	101	00111
15	1111	10001	10101
16		1011	11101

The source decoder for the code constructs an identical copy of the dictionary at the receiving end of the communication system and decodes the received sequence in step with the transmitted data sequence.

It should be observed that the table encoded 44 source bits into 16 code words of 5 bits each, resulting in 80 coded bits. Hence, the algorithm provided no data compression at all. However, the inefficiency is due to the fact that the sequence we have considered is very short. As the sequence is increased in length, the encoding procedure becomes more efficient and results in a compressed sequence at the output of the source.

How do we select the overall length of the table? In general, no matter how large the table is, it will eventually overflow. To solve the overflow problem, the source encoder and source decoder must use an identical procedure to remove phrases from the respective dictionaries that are not useful and substitute new phrases in their place.

The Lempel–Ziv algorithm is widely used in the compression of computer files. The "compress" and "uncompress" utilities under the UNIX© operating system and numerous algorithms under the MS-DOS operating system are implementations of various versions of this algorithm.

6.4
LOSSY DATA COMPRESSION

Our study of data compression techniques thus far has been limited to discrete information sources. For continuous-amplitude information sources, the problem is quite different. For perfect reconstruction of a continuous-amplitude source, the number of required bits is infinite. This is so because representation of a general real number in base 2 requires an infinite number of digits. Therefore, for continuous-amplitude sources lossless compression is impossible, and lossy compression through scalar or vector quantization is employed. In this section we study the notion of lossy data compression and introduce the rate distortion function which provides the fundamental limit on lossy data compression. To introduce the rate distortion function, we need to generalize the notions of entropy and mutual information to continuous random variables.

6.4–1 Entropy and Mutual Information for Continuous Random Variables

The definition of mutual information given for discrete random variables may be extended in a straightforward manner to continuous random variables. In particular, if X and Y are random variables with joint PDF $p(x, y)$ and marginal PDFs $p(x)$ and $p(y)$, the average mutual information between X and Y is defined as

$$I(X; Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} p(x)p(y|x) \log \frac{p(y|x)p(x)}{p(x)p(y)}\, dx\, dy \tag{6.4–1}$$

Although the definition of the average mutual information carries over to continuous random variables, the concept of entropy does not. The problem is that a continuous random variable requires an infinite number of binary digits to represent it exactly. Hence, its self-information is infinite, and, therefore, its entropy is also infinite.

Nevertheless, we shall define a quantity that we call the *differential entropy* of the continuous random variable X as

$$H(X) = -\int_{-\infty}^{\infty} p(x)\log p(x)\,dx \tag{6.4–2}$$

We emphasize that this quantity does *not* have the physical meaning of self-information, although it may appear to be a natural extension of the definition of entropy for a discrete random variable (see Problem 6.15).

By defining the average conditional entropy of X given Y as

$$H(X|Y) = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x, y)\log p(x|y)\,dx\,dy \tag{6.4–3}$$

the average mutual information may be expressed as

$$I(X; Y) = H(X) - H(X|Y) \tag{6.4–4}$$

or, alternatively, as

$$I(X; Y) = H(Y) - H(Y|X) \tag{6.4–5}$$

In some cases of practical interest, the random variable X is discrete and Y is continuous. To be specific, suppose that X has possible outcomes x_i, $i = 1, 2, \ldots, n$, and Y is described by its marginal PDF $p(y)$. When X and Y are statistically dependent, we may express $p(y)$ as

$$p(y) = \sum_{i=1}^{n} p(y|x_i)\,\mathrm{P}[x_i] \tag{6.4–6}$$

The mutual information provided about the event $X = x_i$ by the occurrence of the event $Y = y$ is

$$\begin{aligned} I(x_i; y) &= \log\frac{p(y|x_i)\,\mathrm{P}[x_i]}{p(y)\,\mathrm{P}[x_i]} \\ &= \log\frac{p(y|x_i)}{p(y)} \end{aligned} \tag{6.4–7}$$

Then the average mutual information between X and Y is

$$I(X; Y) = \sum_{i=1}^{n}\int_{-\infty}^{\infty} p(y|x_i)\,\mathrm{P}[x_i]\log\frac{p(y|x_i)}{p(y)}\,dy \tag{6.4–8}$$

EXAMPLE 6.4–1. Suppose that X is a discrete random variable with two equally probable outcomes $x_1 = A$ and $x_2 = -A$. Let the conditional PDFs $p(y|x_i)$, $i = 1, 2$, be Gaussian with mean x_i and variance σ^2. That is,

$$\begin{aligned} p(y|A) &= \frac{1}{\sqrt{2\pi}\,\sigma}e^{-(y-A)^2/2\sigma^2} \\ p(y|-A) &= \frac{1}{\sqrt{2\pi}\,\sigma}e^{-(y+A)^2/2\sigma^2} \end{aligned} \tag{6.4–9}$$

The average mutual information obtained from Equation 6.4–8 becomes

$$I(X;Y) = \frac{1}{2}\int_{-\infty}^{\infty}\left[p(y|A)\log\frac{p(y|A)}{p(y)} + p(y|-A)\log\frac{p(y|-A)}{p(y)}\right]dy \qquad (6.4\text{–}10)$$

where

$$p(y) = \frac{1}{2}[p(y|A) + p(y|-A)] \qquad (6.4\text{–}11)$$

Later in this chapter it will be shown that the average mutual information $I(X;Y)$ given by Equation 6.4–10 represents the channel capacity of a binary-input additive white Gaussian noise channel.

6.4–2 The Rate Distortion Function

An analog source emits a message waveform $x(t)$ that is a sample function of a stochastic process $X(t)$. When $X(t)$ is a band-limited, stationary stochastic process, the sampling theorem allows us to represent $X(t)$ by a sequence of uniform samples taken at the Nyquist rate.

By applying the sampling theorem, the output of an analog source is converted to an equivalent discrete-time sequence of samples. The samples are then quantized in amplitude and encoded. One type of simple encoding is to represent each discrete amplitude level by a sequence of binary digits. Hence, if we have L levels, we need $R = \log_2 L$ bits per sample if L is a power of 2, or $R = \lfloor \log_2 L \rfloor + 1$ if L is not a power of 2. On the other hand, if the levels are not equally probable and the probabilities of the output levels are known, we may use Huffman coding to improve the efficiency of the encoding process.

Quantization of the amplitudes of the sampled signal results in data compression, but it also introduces some distortion of the waveform or a loss of signal fidelity. The minimization of this distortion is considered in this section. Many of the results given in this section apply directly to a discrete-time, continuous-amplitude, memoryless Gaussian source. Such a source serves as a good model for the residual error in a number of source coding methods.

In this section we study only the fundamental limits on lossy source coding given by the rate distortion function. Specific techniques to achieve the bounds predicted by theory are not covered in this book. The interested reader is referred to books and papers on scalar and vector quantization, data compression, waveform, audio and video coding referenced at the end of this chapter.

We begin by studying the distortion introduced when the samples from the information source are quantized to a fixed number of bits. By the term *distortion*, we mean some measure of the difference between the actual source samples $\{x_k\}$ and the corresponding quantized values $\{\hat{x}_k\}$ which we denote by $d(x_k, \hat{x}_k)$. For example, a commonly used distortion measure is the squared-error distortion, defined as

$$d(x_k, \hat{x}_k) = (x_k - \hat{x}_k)^2 \qquad (6.4\text{–}12)$$

If $d(x_k, \hat{x}_k)$ is the distortion measure per letter, the distortion between a sequence of n samples $\boldsymbol{x}_n$ and the corresponding n quantized values $\hat{\boldsymbol{x}}_n$ is the average over the n

source output samples, i.e.,

$$d(\boldsymbol{x}_n, \hat{\boldsymbol{x}}_n) = \frac{1}{n} \sum_{k=1}^{n} d(x_k, \hat{x}_k) \tag{6.4–13}$$

The source output is a random process, and hence the n samples in $\boldsymbol{X}_n$ are random variables. Therefore, $d(\boldsymbol{X}_n, \hat{\boldsymbol{X}}_n)$ is a random variable. Its expected value is defined as the distortion D, i.e.,

$$D = \mathrm{E}\big[d(\boldsymbol{X}_n, \hat{\boldsymbol{X}}_n)\big] = \frac{1}{n} \sum_{k=1}^{n} \mathrm{E}\big[d(X_k, \hat{X}_k)\big] = \mathrm{E}\big[d(X, \hat{X})\big] \tag{6.4–14}$$

where the last step follows from the assumption that the source output process is stationary.

Now suppose we have a memoryless source with a continuous-amplitude output X that has a PDF $p(x)$, a quantized amplitude output alphabet $\hat{X}$, and a per letter distortion measure $d(x, \hat{x})$. Then the minimum rate in bits per source output that is required to represent the output X of the memoryless source with a distortion less than or equal to D is called the *rate distortion function* $R(D)$ and is defined as

$$R(D) = \min_{p(\hat{x}|x):\mathrm{E}[d(X,\hat{X})]\leq D} I(X; \hat{X}) \tag{6.4–15}$$

where $I(X; \hat{X})$ is the mutual information between X and $\hat{X}$. In general, the rate $R(D)$ decreases as D increases, or conversely $R(D)$ increases as D decreases.

As seen from the definition of the rate distortion function, $R(D)$ depends on the statistics of the source $p(x)$ as well as the distortion measure $d(x, \hat{x})$. A change in either of these two would change $R(D)$. We also mention here that for many source statistics and distortion measures there exists no closed form for the rate distortion function $R(D)$.

The rate distortion function $R(D)$ of a source is associated with the following fundamental source coding theorem in information theory.

> **SHANNON'S THIRD THEOREM [SOURCE CODING WITH A FIDELITY CRITERION—SHANNON (1959)]** A memoryless source X can be encoded at rate R for a distortion not exceeding D if $R > R(D)$. Conversely, for any code with rate $R < R(D)$ the distortion exceeds D.

It is clear, therefore, that the rate distortion function $R(D)$ for any source represents a lower bound on the source rate that is possible for a given level of distortion.

The Rate Distortion Function for a Gaussian Source with Squared-Error Distortion

One interesting model of a continuous-amplitude, memoryless information source is the Gaussian source model. For this source statistics and squared-error distortion measure $d(x, \hat{x}) = (x - \hat{x})^2$, the rate distortion function is known and is given by

$$R_g(D) = \begin{cases} \frac{1}{2}\log\frac{\sigma^2}{D} & 0 \leq D \leq \sigma^2 \\ 0 & D > \sigma^2 \end{cases} \tag{6.4–16}$$

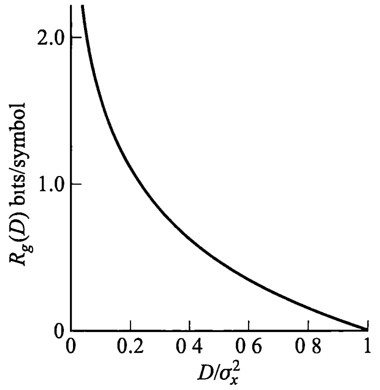


FIGURE 6.4–1
Rate distortion function for a continuous-amplitude, memoryless Gaussian source.

where σ^2 is the variance of the source. Note that $R_g(D)$ is independent of the mean $\mathrm{E}[X]$ of the source. This function is plotted in Figure 6.4–1.

We should note that Equation 6.4–16 implies that no information need be transmitted when the distortion $D \geq \sigma^2$. Specifically, $D = \sigma^2$ can be obtained by using $m = \mathrm{E}[X]$ in the reconstruction of the signal.

If in Equation 6.4–16 we reverse the functional dependence between D and R, we may express D in terms of R as

$$D_g(R) = 2^{-2R}\sigma^2 \tag{6.4–17}$$

This function is called the *distortion rate function* for the discrete-time, memoryless Gaussian source.

When we express the distortion in Equation 6.4–17 in decibels, we obtain

$$10 \log D_g(R) = -6R + 10 \log \sigma^2 \tag{6.4–18}$$

Note that the mean square error distortion decreases at the rate of 6 dB/bit.

Explicit results on the rate distortion functions for general memoryless non-Gaussian sources are not available. However, there are useful upper and lower bounds on the rate distortion function for any discrete-time, continuous-amplitude, memoryless source. An upper bound is given by the following theorem.

THEOREM: UPPER BOUND ON $R(D)$ The rate distortion function of a memoryless, continuous-amplitude source with zero mean and finite variance $\sigma^2$with respect to the mean square error distortion measure is upper-bounded as

$$R(D) \leq \frac{1}{2} \log_2 \frac{\sigma^2}{D}, \qquad 0 \leq D \leq \sigma_x^2 \tag{6.4–19}$$

A proof of this theorem is given by Berger (1971). It implies that the Gaussian source requires the maximum rate among all other sources with the same variance for a specified level of mean square error distortion. Thus the rate distortion function $R(D)$ of any continuous-amplitude memoryless source with finite variance σ^2 satisfies $R(D) \leq R_g(D)$. Similarly, the distortion rate function of the same source satisfies the condition

$$D(R) \leq D_g(R) = 2^{-2R}\sigma^2 \tag{6.4–20}$$

A lower bound on the rate distortion function also exists. This is called the *Shannon lower bound* for a mean square error distortion measure and is given as

$$R^*(D) = H(X) - \frac{1}{2}\log_2 2\pi e D \tag{6.4–21}$$

where $H(X)$ is the differential entropy of the continuous-amplitude, memoryless source. The distortion rate function corresponding to Equation 6.4–21 is

$$D^*(R) = \frac{1}{2\pi e}2^{-2[R-H(X)]} \tag{6.4–22}$$

Therefore, the rate distortion function for any continuous-amplitude, memoryless source is bounded from above and below as

$$R^*(D) \le R(D) \le R_g(D) \tag{6.4–23}$$

and the corresponding distortion rate function is bounded as

$$D^*(R) \le D(R) \le D_g(R) \tag{6.4–24}$$

The differential entropy of the memoryless Gaussian source is

$$H_g(X) = \frac{1}{2}\log_2 2\pi e\sigma^2 \tag{6.4–25}$$

so that the lower bound $R^*(D)$ in Equation 6.4–21 reduces to $R_g(D)$. Now, if we express $D^*(R)$ in terms of decibels and normalize it by setting $\sigma^2 = 1$ (or dividing $D^*(R)$ by σ^2), we obtain from Equation 6.4–22

$$10\log D^*(R) = -6R - 6[H_g(X) - H(X)] \tag{6.4–26}$$

or, equivalently,

$$\begin{aligned} 10\log\frac{D_g(R)}{D^*(R)} &= 6[H_g(X) - H(X)] \quad \text{dB} \\ &= 6[R_g(D) - R^*(D)] \quad \text{dB} \end{aligned} \tag{6.4–27}$$

The relations in Equations 6.4–26 and 6.4–27 allow us to compare the lower bound in the distortion with the upper bound which is the distortion for the Gaussian source. We note that $D^*(R)$ also decreases at -6 dB/bit. We should also mention that the differential entropy $H(X)$ is upper-bounded by $H_g(X)$, as shown by Shannon (1948b).

Rate Distortion Function for a Binary Source with Hamming Distortion

Another interesting and useful case in which a closed-form expression for the rate distortion function exists is the case of a binary source with $p = \mathrm{P}[X = 1] = 1 - \mathrm{P}[X = 0]$. From the lossless source coding theorem, we know that this source can be compressed at any rate R that satisfies $R > H(X) = H_b(p)$ and can be recovered perfectly from the compressed data. However if the rate falls below $H_b(p)$, errors will occur in compression of this source. A measure of distortion that represents the error probability is the *Hamming distortion*, defined as

$$d(x, \hat{x}) = \begin{cases} 1 & x \neq \hat{x} \\ 0 & x = \hat{x} \end{cases} \tag{6.4–28}$$

The average distortion, when this distortion measure is used, is given by

$$\begin{aligned} \mathrm{E}\left[d(X, \hat{X})\right] &= 1 \times \mathrm{P}\left[X \neq \hat{X}\right] + 0 \times \mathrm{P}\left[X = \hat{X}\right] \\ &= \mathrm{P}\left[X \neq \hat{X}\right] \\ &= P_e \end{aligned} \tag{6.4–29}$$

It is seen that the average of Hamming distortion is the error probability in reconstruction of the source.

The rate distortion function for a binary source and with Hamming distortion is given by

$$R(D) = \begin{cases} H_b(p) - H_b(D) & 0 \leq D \leq \min\{p, 1-p\} \\ 0 & \text{otherwise} \end{cases} \tag{6.4–30}$$

Note that as $D \to 0$, we have $R(D) \to H_b(p)$ as expected.

EXAMPLE 6.4–2. A binary symmetric source is to be compressed at a rate of 0.75 bit per source output. For a binary symmetric source we have $p = \frac{1}{2}$ and $H_b(p) = 1$. Since the compression rate, 0.75, is lower than the source entropy, error-free compression is impossible and the best error probability is found by solving $R(D) = 0.75$, where D is P_e because we employ the Hamming distortion. From Equation 6.4–30 we have $R(P_e) = H_b(p) - H_b(P_e) = 1 - H_b(P_e) = 0.75$. Therefore, $H_b(P_e) = 1 - 0.75 = 0.25$, from which we have $P_e = 0.04169$. This is the minimum error probability that can be achieved using a system of unlimited complexity and delay.

6.5

CHANNEL MODELS AND CHANNEL CAPACITY

In the model of a digital communication system described in Chapter 1, we recall that the transmitter building blocks consist of the discrete-input, discrete-output channel encoder followed by the modulator. The function of the discrete channel encoder is to introduce, in a controlled manner, some redundancy in the binary information sequence, which can be used at the receiver to overcome the effects of noise and interference encountered in the transmission of the signal through the channel. The encoding process generally involves taking k information bits at a time and mapping each k-bit sequence into a unique n-bit sequence, called a *codeword*. The amount of redundancy introduced by the encoding of the data in this manner is measured by the ratio n/k. The reciprocal of the ratio, namely k/n, is called the *code rate* and denoted by R_c.

The binary sequence at the output of the channel encoder is fed to the modulator, which serves as the interface to the communication channel. As we have discussed, the modulator may simply map each binary digit into one of two possible waveforms; i.e., a 0 is mapped into $s_1(t)$ and a 1 is mapped into $s_2(t)$. Alternatively, the modulator may transmit q-bit blocks at a time by using $M = 2^q$ possible waveforms.

At the receiving end of the digital communication system, the demodulator processes the channel-corrupted waveform and reduces each waveform to a scalar or a vector that represents an estimate of the transmitted data symbol (binary or M-ary). The detector, which follows the demodulator, may decide whether the transmitted bit is a 0 or a 1. In such a case, the detector has made a hard decision. If we view the decision process at the detector as a form of quantization, we observe that a hard decision corresponds to binary quantization of the demodulator output. More generally, we may consider a detector that quantizes to $Q > 2$ levels, i.e., a Q-ary detector. If M-ary signals are used, then $Q \geq M$. In the extreme case when no quantization is performed, $Q = \infty$. In the case where $Q > M$, we say that the detector has made a soft decision.

The quantized output from the detector is then fed to the channel decoder, which exploits the available redundancy to correct for channel disturbances.

In the following sections, we describe three channel models that will be used to establish the maximum achievable bit rate for the channel.

6.5–1 Channel Models

In this section we describe channel models that will be useful in the design of codes. A general communication channel is described in terms of its set of possible inputs, denoted by $\mathscr{X}$ and called the *input alphabet*; the set of possible channel outputs, denoted by $\mathscr{Y}$ and called the *output alphabet*; and the conditional probability that relates the input and output sequences of any length n, which is denoted by $\mathrm{P}[y_1, y_2, \ldots, y_n \,|x_1, x_2, \ldots, x_n]$, where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ represent input and output sequences of length n, respectively. A channel is called *memoryless* if we have

$$\mathrm{P}[\boldsymbol{y}\,|\boldsymbol{x}] = \prod_{i=1}^{n} \mathrm{P}[y_i\,|x_i] \qquad \text{for all } n \tag{6.5–1}$$

In other words, a channel is memoryless if the output at time i depends only on the input at time i.

The simplest channel model is the binary symmetric channel, which corresponds to the case with $\mathscr{X} = \mathscr{Y} = \{0, 1\}$. This is an appropriate channel model for binary modulation and hard decisions at the detector.

The Binary Symmetric Channel (BSC) Model

Let us consider an additive noise channel and let the modulator and the demodulator/detector be included as parts of the channel. If the modulator employs binary waveforms and the detector makes hard decisions, then the composite channel, shown in Figure 6.5–1, has a discrete-time binary input sequence and a discrete-time binary output sequence. Such a composite channel is characterized by the set $\mathscr{X} = \{0, 1\}$ of

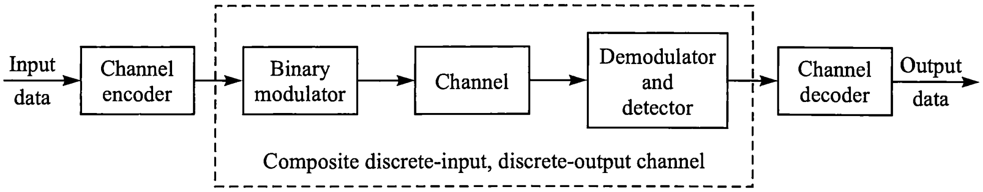


FIGURE 6.5–1
A composite discrete input, discrete output channel formed by including the modulator and the demodulator as part of the channel.

possible inputs, the set of $\mathscr{Y} = \{0, 1\}$ of possible outputs, and a set of conditional probabilities that relate the possible outputs to the possible inputs. If the channel noise and other disturbances cause statistically independent errors in the transmitted binary sequence with average probability p, then

$$\begin{aligned} \mathrm{P}[Y = 0 \,|\, X = 1] &= \mathrm{P}[Y = 1 \,|\, X = 0] = p \\ \mathrm{P}[Y = 1 \,|\, X = 1] &= \mathrm{P}[Y = 0 \,|\, X = 0] = 1 - p \end{aligned} \tag{6.5–2}$$

Thus, we have reduced the cascade of the binary modulator, the waveform channel, and the binary demodulator and detector to an equivalent discrete-time channel which is represented by the diagram shown in Figure 6.5–2. This binary input, binary output, symmetric channel is simply called a binary symmetric channel (BSC). Since each output bit from the channel depends only on the corresponding input bit, we say that the channel is memoryless.

The Discrete Memoryless Channel (DMC)

The BSC is a special case of a more general discrete input, discrete output channel. The discrete memoryless channel is a channel model in which the input and output alphabets $\mathscr{X}$ and $\mathscr{Y}$ are discrete sets and the channel is memoryless. For instance, this is the case when the channel uses an M-ary memoryless modulation scheme and the output of the detector consists of Q-ary symbols. The composite channel consists of modulator-channel-detector as shown in Figure 6.5–1, and its input-output characteristics are described by a set of MQ conditional probabilities

$$\mathrm{P}[y \,|\, x] \qquad \text{for } x \in \mathscr{X}, y \in \mathscr{Y} \tag{6.5–3}$$

The graphical representation of a DMC is shown in Figure 6.5–3.

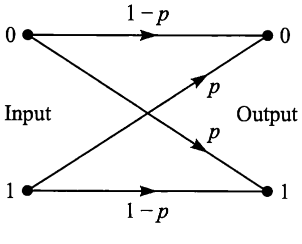


FIGURE 6.5–2
Binary symmetric channel.

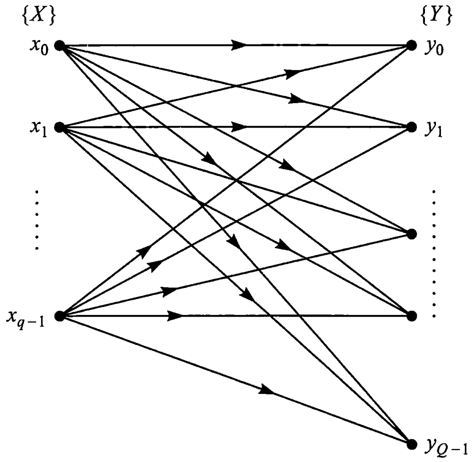


FIGURE 6.5–3
Discrete memoryless channel.

In general, the conditional probabilities $\{\mathrm{P}[y\,|x]\}$ that characterize a DMC can be arranged in an $|\mathscr{X}| \times |\mathscr{Y}|$ matrix of the form $\boldsymbol{P} = [p_{ij}], 1 \le i \le |\mathscr{X}|, 1 \le j \le |\mathscr{Y}|$. $\boldsymbol{P}$ is called the *probability transition matrix* for the channel.

The Discrete-Input, Continuous-Output Channel

Now, suppose that the input to the modulator comprises symbols selected from a finite and discrete input alphabet $\mathscr{X}$, with $|\mathscr{X}| = M$, and the output of the detector is unquantized, i.e., $\mathscr{Y} = \mathbb{R}$. This leads us to define a composite discrete-time memoryless channel that is characterized by the discrete input X, the continuous output Y, and the set of conditional probability density functions

$$p(y|x), \qquad x \in \mathscr{X}, y \in \mathbb{R} \tag{6.5–4}$$

The most important channel of this type is the additive white Gaussian noise (AWGN) channel, for which

$$Y = X + N \tag{6.5–5}$$

where N is a zero-mean Gaussian random variable with variance σ^2. For a given $X = x$, it follows that Y is Gaussian with mean x and variance σ^2. That is,

$$p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x)^2}{2\sigma^2}} \tag{6.5–6}$$

For any given input sequence X_i, $i = 1, 2, \ldots, n$, there is a corresponding output sequence

$$Y_i = X_i + N_i, \qquad i = 1, 2, \ldots, n \tag{6.5–7}$$

The condition that the channel is memoryless may be expressed as

$$p(y_1, y_2, \ldots, y_n|x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p(y_i|x_i) \tag{6.5–8}$$

The Discrete-Time AWGN Channel

This is a channel in which $\mathscr{X} = \mathscr{Y} = \mathbb{R}$. At each instant of time i, an input $x_i \in \mathbb{R}$ is transmitted over the channel. The received symbol is given by

$$y_i = x_i + n_i \tag{6.5–9}$$

where n_i's are iid zero-mean Gaussian random variables with variance σ^2. In addition, it is usually assumed that the channel input satisfies a power constraint of the form

$$\mathrm{E}[X^2] \le P \tag{6.5–10}$$

Under this input power constraint, for any input sequence of the form $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, where n is large with probability approaching 1, we have

$$\frac{1}{n}\sum_{i=1}^{n} x_i^2 = \frac{1}{n}\|\boldsymbol{x}\|^2 \le P \tag{6.5–11}$$

The geometric interpretation of the above constraint is that the input sequences to the channel are inside an n-dimensional sphere of radius $\sqrt{nP}$ centered at the origin.

The AWGN Waveform Channel

We may separate the modulator and the demodulator from the physical channel, and we consider a channel model in which the inputs are waveforms and the outputs are waveforms. Let us assume that such a channel has a given bandwidth W, with ideal frequency response $C(f) = 1$ within the frequency range $[-W, +W]$, and the signal at its output is corrupted by additive white Gaussian noise. Suppose that $x(t)$ is a band-limited input to such a channel and $y(t)$ is the corresponding output. Then

$$y(t) = x(t) + n(t) \tag{6.5–12}$$

where $n(t)$ represents a sample function of the additive white Gaussian noise process with power spectral density of $\frac{N_0}{2}$. Usually, the channel input is subject to a power constraint of the form

$$\mathrm{E}[X^2(t)] \le P \tag{6.5–13}$$

which for ergodic inputs results in an input power constraint of the form

$$\lim_{T\to\infty} \frac{1}{T}\int_{-T/2}^{T/2} x^2(t)\,dt \le P \tag{6.5–14}$$

A suitable method for defining a set of probabilities that characterize the channel is to expand $x(t)$, $y(t)$, and $n(t)$ into a complete set of orthonormal functions. From the dimensionality theorem discussed in Section 4.6–1, we know that the dimensionality of the space of signals with an approximate bandwidth of W and an approximate duration of T is roughly $2WT$. Therefore we need a set of $2W$ dimensions per second to expand the input signals. We can add adequate signals to this set to make it a complete set of orthonormal signals that, by Example 2.8–1, can be used for expansion of white processes. Hence, we can express $x(t)$, $y(t)$, and $n(t)$ in the form

$$\begin{aligned} x(t) &= \sum_j x_j \phi_j(t) \\ n(t) &= \sum_j n_j \phi_j(t) \\ y(t) &= \sum_j y_j \phi_j(t) \end{aligned} \tag{6.5–15}$$

where $\{y_j\}$, $\{x_j\}$, and $\{n_j\}$ are the sets of coefficients in the corresponding expansions, e.g.,

$$\begin{aligned} y_j &= \int_{-\infty}^{\infty} y(t)\phi_j(t)\,dt \\ &= \int_{-\infty}^{\infty} (x(t)+n(t))\,\phi_j(t)\,dt \\ &= x_j + n_j \end{aligned} \tag{6.5–16}$$

We may now use the coefficients in the expansion for characterizing the channel. Since

$$y_j = x_j + n_j \tag{6.5–17}$$

where n_j's are iid zero-mean Gaussian random variables with variance $\sigma^2 = \frac{N_0}{2}$, it follows that

$$p(y_j|x_j) = \frac{1}{\sqrt{\pi N_0}} e^{-\frac{(y_j-x_j)^2}{N_0}}, \qquad i = 1, 2, \ldots \tag{6.5–18}$$

and by the independence of n_j's

$$p(y_1, y_2, \ldots, y_N | x_1, x_2, \ldots, x_N) = \prod_{j=1}^{N} p(y_j|x_j) \tag{6.5–19}$$

for any N. In this manner, the AWGN waveform channel is reduced to an equivalent discrete-time channel characterized by the conditional PDF given in Equation 6.5–18. The power constraint on the input waveforms given by Equation 6.5–14 can be written as

$$\begin{aligned} \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x^2(t)\,dt &= \lim_{T\to\infty} \frac{1}{T} \sum_{j=1}^{2WT} x_j^2 \\ &= \lim_{T\to\infty} \frac{1}{T} \times 2WT\mathrm{E}\left[X^2\right] \\ &= 2W\mathrm{E}\left[X^2\right] \\ &\le P \end{aligned} \tag{6.5–20}$$

where the first equality follows from orthonormality of the $\{\phi_j(t),\ j = 1, 2, \ldots, 2WT\}$, the second equality follows from the law of large numbers applied to the sequence

$\{x_j, 1 \leq j \leq 2WT\}$, and the last inequality follows from Equation 6.5–14. From Equation 6.5–20 we conclude that in the discrete-time channel model we have

$$E[X^2] \leq 2\frac{P}{W} \tag{6.5–21}$$

From Equations 6.5–19 and 6.5–21 it is clear that the waveform AWGN channel with bandwidth constraint W and input power constraint P is equivalent with $2W$ uses per second of a discrete-time AWGN channel with noise variance of $\sigma^2 = \frac{N_0}{2}$ and an input power constraint given by Equation 6.5–21.

6.5–2 Channel Capacity

We have seen that the entropy and the rate distortion function provide the fundamental limits for lossless and lossy data compression. The entropy and the rate distortion function provide the minimum required rates for compression of a discrete memoryless source subject to the condition that it can be losslessly recovered, or can be recovered with a distortion not exceeding a specific D, respectively. In this section we introduce a third fundamental quantity called *channel capacity* that provides the maximum rate at which reliable communication over a channel is possible.

Let us consider a discrete memoryless channel with crossover probability of p. In transmission of 1 bit over this channel the error probability is p, and when a sequence of length n is transmitted over this channel, the probability of receiving the sequence correctly is $(1-p)^n$ which goes to zero as $n \to \infty$. One approach to improve the performance of this channel is not to use all binary sequences of length n as possible inputs to this channel but to choose a subset of them and use only that subset. Of course this subset has to be selected in such a way that the sequences in it are in some sense "far apart" such that they can be recognized and correctly detected at the receiver even in the presence of channel errors.

Let us assume a binary sequence of length n is transmitted over the channel. If n is large, the law of large numbers states that with high probability np bits will be received in error, and as $n \to \infty$, the probability of receiving np bits in error approaches 1. The number of sequences of length n that are different from the transmitted sequence at np positions (np an integer) is

$$\binom{n}{np} = \frac{n!}{(np)!(n(1-p))!} \tag{6.5–22}$$

By using Stirling's approximation that states for large n we have

$$n! \approx \sqrt{2\pi n}\, n^n e^{-n} \tag{6.5–23}$$

Equation 6.5–22 can be approximated as

$$\binom{n}{np} \approx 2^{nH_b(p)} \tag{6.5–24}$$

This means that when any sequence of length n is transmitted, it is highly probable that one of the $2^{nH_b(p)}$ that are different from the transmitted sequence in np positions will be received. If we insist on using all possible input sequences for this channel, errors are inevitable since there will be considerable overlap between the received sequences. However, if we use a subset of all possible input sequences, and choose this subset such that the set of highly probable received sequences for each element of this subset is nonoverlapping, then reliable communication is possible. Since the total number of binary sequences of length n at the channel output is 2^n, we can have at most

$$M = \frac{2^n}{2^{nH_b(p)}} = 2^{n(1-H_b(p))} \tag{6.5–25}$$

sequences of length n transmitted without their corresponding highly probable received sequences overlapping. Therefore, in n uses of the channel we can transmit M messages, and the rate, i.e., the information transmitted per each use of the channel, is given by

$$R = \frac{1}{n}\log_2 M = 1 - H_b(p) \tag{6.5–26}$$

The quantity $1 - H_b(p)$ is the maximum rate for reliable communication over a binary symmetric channel and is called the *capacity* of this channel. In general the capacity of a channel, denoted by C, is the maximum rate at which *reliable communication*, i.e., communication with arbitrary small error probability, over the channel is possible.

For an arbitrary DMC the capacity is given by

$$C = \max_{\boldsymbol{p}} I(X; Y) \tag{6.5–27}$$

where the maximization is over all PMFs of the form $\boldsymbol{p} = \left(p_1, p_2, \ldots, p_{|\mathscr{X}|}\right)$ on the input alphabet $\mathscr{X}$. The p_i's naturally satisfy the constraints

$$\begin{aligned} &p_i \geq 0 \qquad i = 1, 2, \ldots, |\mathscr{X}| \\ &\sum_{i=1}^{|\mathscr{X}|} p_i = 1 \end{aligned} \tag{6.5–28}$$

The units of C are *bits per transmission* or *bits per channel use*, if in computing $I(X; Y)$ logarithms are in base 2, and nats per transmission when the natural logarithm (base e) is used. If a symbol enters the channel every τ_s seconds, the channel capacity is C/τ_s bits/s or nats/s.

The significance of the channel capacity is due to the following fundamental theorem, known as the *noisy channel coding theorem.*

SHANNON'S SECOND THEOREM—THE NOISY CHANNEL CODING THEOREM (SHANNON 1948) Reliable communication over a discrete memoryless channel is possible if the communication rate R satisfies $R < C$, where C is the channel capacity. At rates higher than capacity, reliable communication is impossible.

The noisy channel coding theorem is of utmost significance in communication theory. This theorem expresses the limit to reliable communication and provides a yardstick to measure the performance of communication systems. A system performing near capacity is a near optimal system and does not have much room for improvement. On the other hand a system operating far from this fundamental bound can be improved mainly through coding techniques described in Chapters 7 and 8. Although we have stated the noisy channel coding theorem for discrete memoryless channels, this theorem applies to a much larger class of channels. For details see the paper by Verdu and Han (1994).

We also note that Shannon's proof of the noisy channel coding theorem is nonconstructive and employs a technique introduced by Shannon called *random coding*. In this technique instead of looking for the best possible coding scheme and analyzing its performance, which is a difficult task, all possible coding schemes are considered and the performance of the system is averaged over them. Then it is proved that if $R < C$, the average error probability tends to zero. This proves that among all possible coding schemes there exists at least one code for which the error probability tends to zero. We will discuss this notion in greater detail in Section 6.8–2.

EXAMPLE 6.5–1. For a BSC, due to the symmetry of the channel, the capacity is achieved for a uniform input distribution, i.e., for $\mathrm{P}[X = 1] = \mathrm{P}[X = 0] = \frac{1}{2}$. The maximum mutual information is given by

$$C = 1 + p \log 2p + (1 - p) \log 2(1 - p) = 1 - H(p) \tag{6.5–29}$$

This agrees with our earlier intuitive reasoning. A plot of C versus p is illustrated in Figure 6.5–4. Note that for $p = 0$, the capacity is 1 bit/channel use. On the other hand, for $p = \frac{1}{2}$, the mutual information between input and output is zero. Hence, the channel capacity is zero. For $\frac{1}{2} < p \leq 1$, we may reverse the position of 0 and 1 at the output of the BSC, so that C becomes symmetric with respect to the point $p = \frac{1}{2}$. In our treatment of binary modulation and demodulation given in Chapter 4, we showed that p is a monotonic function of the SNR per bit. Consequently when C is plotted as a function of the SNR per bit, it increases monotonically as the SNR per bit increases. This characteristic behavior of C versus SNR per bit is illustrated in Figure 6.5–5 for the case where the binary modulation scheme is antipodal signaling.

The Capacity of the Discrete-Time Binary-Input AWGN Channel We consider the binary-input AWGN channel with inputs $\pm A$ and noise variance σ^2. The transition probability density function for this channel is defined by Equation 6.5–6 where $x = \pm A$. By symmetry, the capacity of this channel is achieved by a symmetric input PMF, i.e., by letting $\mathrm{P}[X = A] = \mathrm{P}[X = -A] = \frac{1}{2}$. Using these input probabilities, the

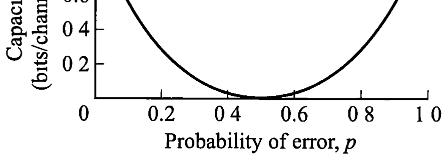


FIGURE 6.5–4
The capacity of a BSC.

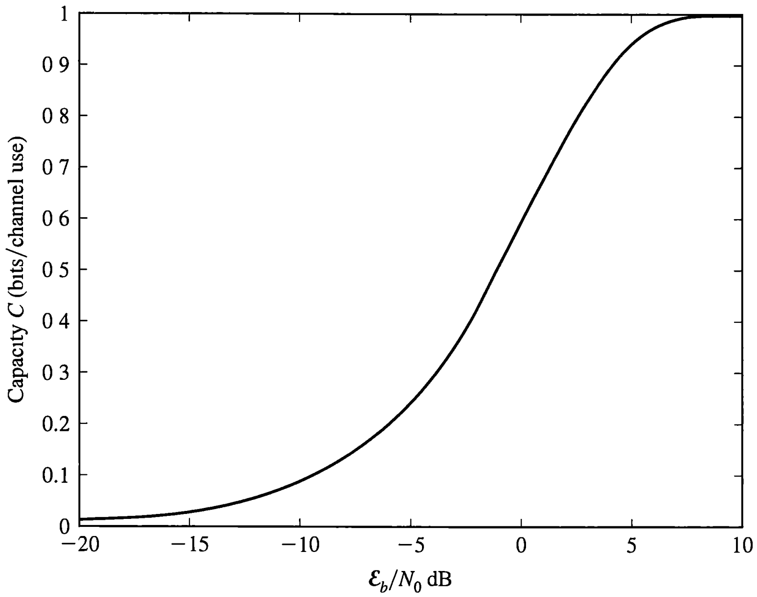


FIGURE 6.5–5
The capacity plot versus SNR per bit.

capacity of this channel in bits per channel use is given by

$$C = \frac{1}{2}\int_{-\infty}^{\infty} p(y|A)\log_2 \frac{p(y|A)}{p(y)}\,dy + \frac{1}{2}\int_{-\infty}^{\infty} p(y|-A)\log_2 \frac{p(y|-A)}{p(y)}\,dy \qquad (6.5\text{–}30)$$

The capacity in this case does not have a closed form. In Problem 6.50 it is shown that the capacity of this channel can be written as

$$C = \frac{1}{2}g\left(\frac{A}{\sigma}\right) + \frac{1}{2}\left(-\frac{A}{\sigma}\right) \qquad (6.5\text{–}31)$$

where

$$g(x) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{(u-x)^2}{2}} \log_2 \frac{2}{1+e^{-2ux}}\,du \qquad (6.5\text{–}32)$$

Figure 6.5–6 illustrates C as a function of the ratio $\frac{\mathcal{E}_b}{N_0}$. Note that C increases monotonically from 0 to 1 bit per symbol as this ratio increases. The two points shown on this plot correspond to transmission rates of $\frac{1}{2}$ and $\frac{1}{3}$. Note that the $\frac{\mathcal{E}_b}{N_0}$ required to achieve these rates is 0.188 and −0.496, respectively.

Capacity of Symmetric Channels It is interesting to note that in the two channel models described above, the BSC and the discrete-time binary-input AWGN channel, the choice of equally probable input symbols maximizes the average mutual information. Thus, the capacity of the channel is obtained when the input symbols are equally probable. This is not always the solution for the capacity formulas given in Equation 6.5–27, however. In the two channel models considered above, the channel transition probabilities exhibit a form of symmetry that results in the maximum of

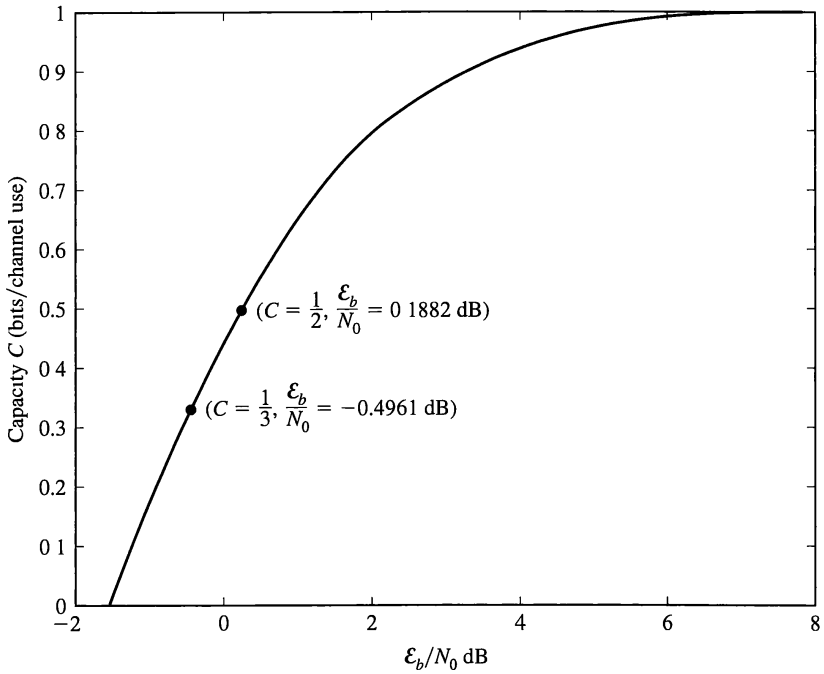


FIGURE 6.5–6
The capacity of binary input AWGN channel.

$I(X; Y)$ being obtained when the input symbols are equally probable. A channel is called a *symmetric channel* when each row of $\boldsymbol{P}$ is a permutation of any other row and each column of it is a permutation of any other column. For symmetric channels, input symbols with equal probability maximize $I(X; Y)$. The resulting capacity of a symmetric channel is

$$C = \log_2 |\mathscr{Y}| - H(\boldsymbol{p}) \tag{6.5–33}$$

where $\boldsymbol{p}$ is the PMF given by any row of $\boldsymbol{P}$. Note that since the rows of $\boldsymbol{P}$ are permutations of each other, the entropy of the PMF corresponding to each row is independent of the row. One example of a symmetric channel is the binary symmetric channel for which $\boldsymbol{p} = (p, 1 - p)$ and $|\mathscr{Y}| = 2$, therefore $C = 1 - H_b(p)$.

In general, for an arbitrary DMC, the necessary and sufficient conditions for the set of input probabilities $\{\mathrm{P}[x]\}$ to maximize $I(X; Y)$ and, thus, to achieve capacity on a DMC are that (Problem 6.52)

$$\begin{aligned} I(x; Y) &= C \qquad \text{for all } x \in \mathscr{X} \text{ with } \mathrm{P}[x] > 0 \\ I(x; Y) &\leq C \qquad \text{for all } x \in \mathscr{X} \text{ with } \mathrm{P}[x] = 0 \end{aligned} \tag{6.5–34}$$

where C is the capacity of the channel and

$$I(x; Y) = \sum_{y \in \mathscr{Y}} \mathrm{P}[y \mid x] \log \frac{\mathrm{P}[y \mid x]}{\mathrm{P}[y]} \tag{6.5–35}$$

Usually, it is relatively easy to check if the equally probable set of input symbols satisfies the conditions given in Equation 6.5–34. If they do not, then one must determine the set of unequal probabilities $\{\mathrm{P}[x]\}$ that satisfies Equation 6.5–34.

The Capacity of Discrete-Time AWGN Channel with an Input Power Constraint Here we deal with the channel model

$$Y_i = X_i + N_i \tag{6.5–36}$$

where N_i's are iid zero-mean Gaussian random variables with variance σ^2 and input X is subject to the power constraint

$$\mathrm{E}[X^2] \leq P \tag{6.5–37}$$

For large n, the law of large numbers states that

$$\frac{1}{n}\|\boldsymbol{y}\|^2 \to \mathrm{E}[X^2] + \mathrm{E}[N^2] \leq P + \sigma^2 \tag{6.5–38}$$

Equation 6.5–38 states that the output vector $\boldsymbol{y}$ is inside an n-dimensional sphere of radius $\sqrt{n(P+\sigma^2)}$. If $\boldsymbol{x}$ is transmitted, the received vector $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{n}$ satisfies

$$\frac{1}{n}\|\boldsymbol{y} - \boldsymbol{x}\|^2 = \frac{1}{n}\|\boldsymbol{n}\|^2 \to \sigma^2 \tag{6.5–39}$$

which means if $\boldsymbol{x}$ is transmitted, with high probability $\boldsymbol{y}$ will be in an n-dimensional sphere of radius $\sqrt{n\sigma^2}$ and centered at $\boldsymbol{x}$. The maximum number of spheres of radius $\sqrt{n\sigma^2}$ that can be packed in a sphere of radius $\sqrt{n(P+\sigma^2)}$ is the ratio of the volumes of the spheres. The volume of an n-dimensional sphere is given by $V_n = B_n R^n$, where B_n is given by Equation 4.7–15. Therefore, the maximum number of messages that can be transmitted and still be resolvable at the receiver is

$$M = \frac{B_n\left(\sqrt{n(P+\sigma^2)}\right)^n}{B_n\left(\sqrt{n\sigma^2}\right)^n} = \left(1 + \frac{P}{\sigma^2}\right)^{\frac{n}{2}} \tag{6.5–40}$$

which results in a rate of

$$R = \frac{1}{n}\log_2 M = \frac{1}{2}\log_2\left(1 + \frac{P}{\sigma^2}\right) \quad \text{bits/transmission} \tag{6.5–41}$$

This result can be obtained by direct maximization of $I(X;Y)$ over all input PDFs $p(x)$ that satisfy the power constraint $\mathrm{E}[X^2] \leq P$. The input PDF that maximizes $I(X;Y)$ is a zero-mean Gaussian PDF with variance P. A plot of the capacity for this channel versus SNR per bit is shown in Figure 6.5–7. The points corresponding to $C = \frac{1}{2}$ and $C = \frac{1}{3}$ are also shown on the figure.

The Capacity of Band-Limited Waveform AWGN Channel with an Input Power Constraint As we have seen by the discussion following Equation 6.5–21, this channel model is equivalent to $2W$ uses per second of a discrete-time AWGN channel with input

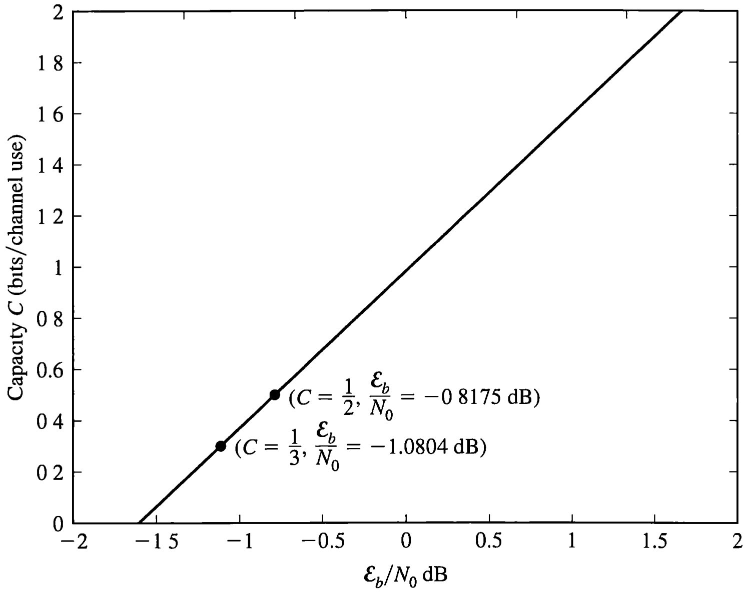


FIGURE 6.5–7
The capacity of a discrete-time AWGN channel.

power constraint of $\frac{P}{2W}$ and noise variance of $\sigma^2 = \frac{N_0}{2}$. The capacity of this discrete-time channel is

$$C = \frac{1}{2}\log_2\left(1 + \frac{\frac{P}{2W}}{\frac{N_0}{2}}\right) = \frac{1}{2}\log_2\left(1 + \frac{P}{N_0 W}\right) \quad \text{bits/channel use} \qquad (6.5\text{–}42)$$

Therefore, the capacity of the continuous-time channel is given by

$$C = 2W \times \frac{1}{2}\log_2\left(1 + \frac{P}{N_0 W}\right) = W\log_2\left(1 + \frac{P}{N_0 W}\right) \quad \text{bits/s} \qquad (6.5\text{–}43)$$

This is the celebrated equation for the capacity of a band-limited AWGN channel with input power constraint derived by Shannon (1948b).

From Equation 6.5–43, it is clear that the capacity increases by increasing P, and in fact $C \to \infty$ as $P \to \infty$. However, the rate by which the capacity increases at large values of P is a logarithmic rate. Increasing W, however, has a dual role on the capacity. On one hand, it causes the capacity to be increased because higher bandwidth means more transmissions over the channel per unit time. On the other hand, increasing W decreases the SNR defined by $\frac{P}{N_0 W}$. This is so because increasing the bandwidth increases the effective noise power entering the receiver. To see how the capacity changes as $W \to \infty$, we need to use the relation $\ln(1 + x) \to x$ as $x \to 0$ to get

$$C_\infty = \lim_{W\to\infty} W\log_2\left(1 + \frac{P}{N_0 W}\right) = \left(\log_2 e\right)\frac{P}{N_0} \approx 1.44\,\frac{P}{N_0} \quad \text{bits/s} \qquad (6.5\text{–}44)$$

It is clear that the having infinite bandwidth cannot increase the capacity indefinitely, and its effect is limited by the amount of available power. This is in contrast to the effect of having infinite power that, regardless of the amount of available bandwidth, can increase the capacity indefinitely.

To derive a fundamental relation between the bandwidth and power efficiency of a communication system, we note that for reliable communication we must have $R < C$ which in the case of a band-limited AWGN channel is given by

$$R < W \log_2 \left(1 + \frac{P}{N_0 W}\right) \tag{6.5–45}$$

Dividing both sides by W and using $r = R/W$, as previously defined in Equation 4.6–1 as the bandwidth efficiency, we obtain

$$r < \log_2 \left(1 + \frac{P}{N_0 W}\right) \tag{6.5–46}$$

Using the relation

$$\mathcal{E}_b = \frac{\mathcal{E}}{\log_2 M} = \frac{PT_s}{\log_2 M} = \frac{P}{R} \tag{6.5–47}$$

we obtain

$$r < \log_2 \left(1 + \frac{\mathcal{E}_b R}{N_0 W}\right) = \log_2 \left(1 + r\frac{\mathcal{E}_b}{N_0}\right) \tag{6.5–48}$$

from which we have

$$\frac{\mathcal{E}_b}{N_0} > \frac{2^r - 1}{r} \tag{6.5–49}$$

This relation states the condition for reliable communication in terms of bandwidth efficiency r and $\frac{\mathcal{E}_b}{N_0}$ which is a measure of power efficiency of a system. A plot of this relation is given in Figure 4.6–1. The minimum value of $\frac{\mathcal{E}_b}{N_0}$ for which reliable communication is possible is obtained by letting $r \to 0$ in Equation 6.5–49, which results in

$$\frac{\mathcal{E}_b}{N_0} > \ln 2 \approx 0.693 \sim -1.6 \text{ dB} \tag{6.5–50}$$

This is the minimum required value of $\frac{\mathcal{E}_b}{N_0}$ for any communication system. No system can transmit reliably below this limit and in order to achieve this limit we need to let $r \to 0$, or equivalently, $W \to \infty$.

6.6
ACHIEVING CHANNEL CAPACITY WITH ORTHOGONAL SIGNALS

In Section 4.4–1, we used a simple union bound to show that, for orthogonal signals, the probability of error can be made as small as desired by increasing the number M of waveforms, provided that $\mathcal{E}_b/N_0 > 2 \ln 2$. We indicated that the simple union bound does not produce the smallest lower bound on the SNR per bit. The problem is that the upper bound used in $Q(x)$ is very loose for small x.

An alternative approach is to use two different upper bounds for $Q(x)$, depending on the value of x. Beginning with Equation 4.4–10 and using the inequality $(1-x)^n \geq 1 - nx$, which holds for $0 \leq x \leq 1$ and $n \geq 1$, we observe that

$$1 - [1 - Q(x)]^{M-1} \leq (M-1)Q(x) < Me^{-x^2/2} \tag{6.6–1}$$

This is just the union bound, which is tight when x is large, i.e., for $x > x_0$, where x_0 depends on M. When x is small, the union bound exceeds unity for large M. Since

$$1 - [1 - Q(x)]^{M-1} \leq 1 \tag{6.6–2}$$

for all x, we may use this bound for $x < x_0$ because it is tighter than the union bound. Thus Equation 4.4–10 may be upper-bounded as

$$P_e < \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x_0} e^{-\left(x-\sqrt{2\gamma}\right)^2/2}\, dx + \frac{M}{\sqrt{2\pi}} \int_{x_0}^{\infty} e^{-x^2/2} e^{-\left(x-\sqrt{2\gamma}\right)^2/2}\, dx \tag{6.6–3}$$

where $\gamma = \frac{\mathcal{E}}{N_0}$.

The value of x_0 that minimizes this upper bound is found by differentiating the right-hand side of Equation 6.6–3 and setting the derivative equal to zero. It is easily verified that the solution is

$$e^{x_0^2/2} = M \tag{6.6–4}$$

or, equivalently,

$$x_0 = \sqrt{2 \ln M} = \sqrt{2 \ln 2 \log_2 M} = \sqrt{2k \ln 2} \tag{6.6–5}$$

Having determined x_0, we now compute simple exponential upper bounds for the integrals in Equation 6.6–3. For the first integral, we have

$$\begin{aligned} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x_0} e^{-\left(x-\sqrt{2\gamma}\right)^2/2}\, dx &= \frac{1}{\sqrt{\pi}} \int_{-\infty}^{-\left(\sqrt{2\gamma}-x_0\right)/\sqrt{2}} e^{-u^2}\, du \\ &= Q\left(\sqrt{2\gamma} - x_0\right), \qquad x_0 \leq \sqrt{2\gamma} \\ &< e^{-\left(\sqrt{2\gamma}-x_0\right)^2/2}, \qquad x_0 \leq \sqrt{2\gamma} \end{aligned} \tag{6.6–6}$$

The second integral is upper-bounded as follows:

$$\begin{aligned} \frac{M}{\sqrt{2\pi}} \int_{x_0}^{\infty} e^{-x^2/2} e^{-\left(x-\sqrt{2\gamma}\right)^2/2}\, dx &= \frac{M}{\sqrt{2\pi}}\, e^{-\gamma/2} \int_{x_0-\sqrt{\gamma/2}}^{\infty} e^{-u^2}\, du \\ &< \begin{cases} Me^{-\gamma/2} & x_0 \leq \sqrt{\gamma/2} \\ Me^{-\gamma/2} e^{-\left(x_0-\sqrt{\gamma/2}\right)^2} & x_0 > \sqrt{\gamma/2} \end{cases} \end{aligned} \tag{6.6–7}$$

Combining the bounds for the two integrals and substituting $e^{x_0^2/2}$ for M, we obtain

$$P_e < \begin{cases} e^{-\left(\sqrt{2\gamma}-x_0\right)^2/2} + e^{(x_0^2-\gamma)/2} & 0 \leq x_0 \leq \sqrt{\gamma/2} \\ e^{-\left(\sqrt{2\gamma}-x_0\right)^2/2} + e^{(x_0^2-\gamma)/2} e^{-\left(x_0-\sqrt{\gamma/2}\right)^2} & \sqrt{\gamma/2} < x_0 \leq \sqrt{2\gamma} \end{cases} \tag{6.6–8}$$

In the range $0 \le x_0 \le \sqrt{\gamma/2}$, the bound may be expressed as

$$P_e < e^{(x_0^2-\gamma)/2}\left(1+e^{-\left(x_0-\sqrt{\gamma/2}\right)^2}\right) < 2e^{(x_0^2-\gamma)/2}, \qquad 0 \le x_0 \le \sqrt{\gamma/2} \quad (6.6\text{–}9)$$

In the range $\sqrt{\gamma/2} \le x_0 \le \sqrt{2\gamma}$, the two terms in Equation 6.6–8 are identical. Hence,

$$P_e < 2e^{-\left(\sqrt{2\gamma}-x_0\right)^2/2}, \qquad \sqrt{\gamma/2} \le x_0 \le \sqrt{2\gamma} \quad (6.6\text{–}10)$$

Now we substitute for x_0 and γ. Since $x_0 = 2\ln M = \sqrt{2k\ln 2}$ and $\gamma = k\gamma_b$, the bounds in Equations 6.6–9 and 6.6–10 may be expressed as

$$P_e < \begin{cases} 2e^{-k(\gamma_b-2\ln 2)/2} & \ln\ M \le \frac{1}{4}\gamma \\ 2e^{-k\left(\sqrt{\gamma_b}-\sqrt{\ln 2}\right)^2} & \frac{1}{4}\gamma \le \ln M \le \gamma \end{cases} \quad (6.6\text{–}11)$$

The first upper bound coincides with the union bound presented earlier, but it is loose for large values of M. The second upper bound is better for large values of M. We note that $P_e \to 0$ as $k \to \infty$ $(M \to \infty)$ provided that $\gamma_b > \ln 2$. But $\ln 2$ is the limiting value of the SNR per bit required for reliable transmission when signaling at a rate equal to the capacity of the infinite-bandwidth AWGN channel, as shown in Equation 6.5–50. In fact, when the substitutions $y_0 = \sqrt{2k\ln 2} = \sqrt{2RT\ln 2}$ and $\gamma = \mathcal{E}/N_0 = TP/N_0 = TC_\infty \ln 2$, which follow from Equation 6.5–44, are made into the two upper bounds given in Equations 6.6–9 and 6.6–10, the result is

$$P_e < \begin{cases} 2\times 2^{-T\left(\frac{1}{2}C_\infty - R\right)} & 0 \le R \le \frac{1}{4}C_\infty \\ 2\times 2^{-T\left(\sqrt{C_\infty}-\sqrt{R}\right)^2} & \frac{1}{4}C_\infty \le R \le C_\infty \end{cases} \quad (6.6\text{–}12)$$

Thus we have expressed the bounds in terms of C_∞ and the bit rate in the channel. The first upper bound is appropriate for rates below $\frac{1}{4}C_\infty$, while the second is tighter than the first for rates between $\frac{1}{4}C_\infty$ and C_∞. Clearly, the probability of error can be made arbitrarily small by making $T \to \infty$ $(M \to \infty$ for fixed $R)$, provided that $R < C_\infty = P/(N_0 \ln 2)$. Furthermore, we observe that the set of orthogonal waveforms achieves the channel capacity bound as $M \to \infty$, when the rate $R < C_\infty$.

6.7
THE CHANNEL RELIABILITY FUNCTION

The exponential bounds on the error probability for M-ary orthogonal signals on an infinite-bandwidth AWGN channel given by Equation 6.6–12 may be expressed as

$$P_e < 2\times 2^{-TE(R)} \quad (6.7\text{–}1)$$

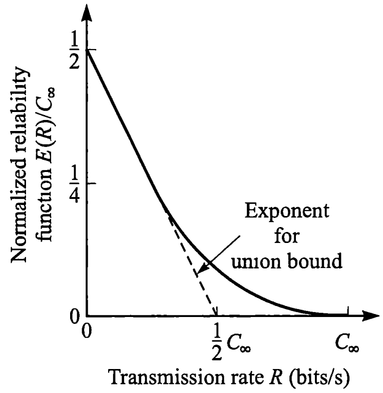


FIGURE 6.7–1
Channel reliability function for the infinite-bandwidth AWGN channel.

The exponential factor

$$E(R) = \begin{cases} \frac{1}{2}C_\infty - R & 0 \le R \le \frac{1}{4}C_\infty \\ \left(\sqrt{C_\infty} - \sqrt{R}\right)^2 & \frac{1}{4}C_\infty \le R \le C_\infty \end{cases} \tag{6.7–2}$$

in Equation 6.7–2 is called the *channel reliability function* for the infinite-bandwidth AWGN channel. A plot of $E(R)/C_\infty$ is shown in Figure 6.7–1. Also shown is the exponential factor for the union bound on P_e, given by Equation 4.4–17, which may be expressed as

$$P_e \le \frac{1}{2} \times 2^{-T(\frac{1}{2}C_\infty - R)}, \qquad 0 \le R \le \frac{1}{2}C_\infty \tag{6.7–3}$$

Clearly, the exponential factor in Equation 6.7–3 is not as tight as $E(R)$, due to the looseness of the union bound.

The bound given by Equations 6.7–1 and 6.7–2 has been shown by Gallager (1965) to be exponentially tight. This means that there does not exist another reliability function, say $E_1(R)$, satisfying the condition $E_1(R) > E(R)$ for any R. Consequently, the error probability is bounded from above and below as

$$K_l 2^{-TE(R)} \le P_e \le K_u 2^{-TE(R)} \tag{6.7–4}$$

where the constants have only a weak dependence on T in the sense that

$$\lim_{T\to\infty} \frac{1}{T} \ln K_l = \lim_{T\to\infty} \frac{1}{T} \ln K_u = 0 \tag{6.7–5}$$

Since orthogonal signals are asymptotically optimal for large M, the lower bound in Equation 6.7–4 applies for any signal set. Hence, the reliability function $E(R)$ given by Equation 6.7–2 determines the exponential characteristics of the error probability for digital signaling over the infinite-bandwidth AWGN channel.

Although we have presented the channel reliability function for the infinite-bandwidth AWGN channel, the notion of channel reliability function can be applied to many channel models. In general, for many channel models, the average error probability over all the possible codes generated randomly satisfies an expression similar to

Equation 6.7–4 of the form

$$K_l 2^{-nE(R)} \leq P_e \leq K_u 2^{-nE(R)} \tag{6.7–6}$$

where $E(R)$ is positive for all $R < C$. Therefore, if $R < C$, it is possible to arbitrarily decrease the error probability by increasing n. This, of course, requires unlimited decoding complexity and delay. The exact expression for the channel reliability function can be derived for just a few channel models. For more details on the channel reliability function, the interested reader is referred to the book by Gallager (1968).

Although the error probability can be made small by increasing the number of orthogonal, biorthogonal, or simplex signals, with $R < C_\infty$, for a relatively modest number of signals, there is a large gap between the actual performance and the best achievable performance given by the channel capacity formula. For example, from Figure 4.6–1, we observe that a set of $M = 16$ orthogonal signals detected coherently requires an SNR per bit of approximately 7.5 dB, to achieve a bit error rate of $P_e = 10^{-5}$. In contrast, the channel capacity formula indicates that for a $C/W = 0.5$, reliable transmission is possible with an SNR of -0.8 dB, as indicated in Figure 6.5–7. This represents a rather large difference of 8.3 dB/bit and serves as a motivation for searching for more efficient signaling waveforms. In this chapter and in Chapters 7 and 8, we demonstrate that coded waveforms can reduce this gap considerably.

Similar gaps in performance also exist in the bandwidth-limited region of Figure 4.6–1, where $R/W > 1$. In this region, however, we must be more clever in how we use coding to improve performance, because we cannot expand the bandwidth as in the power-limited region. The use of coding techniques for bandwidth-efficient communication is treated in Chapters 7 and 8.

6.8
THE CHANNEL CUTOFF RATE

The design of coded modulation for efficient transmission of information may be divided into two basic approaches. One is the algebraic approach, which is primarily concerned with the design of coding and decoding techniques for specific classes of codes, such as cyclic block codes and convolutional codes. The second is the probabilistic approach, which is concerned with the analysis of the performance of a general class of coded signals. This approach yields bounds on the probability of error that can be attained for communication over a channel having some specified characteristic.

In this section, we adopt the probabilistic approach to coded modulation. The algebraic approach, based on block codes and on convolutional codes, is treated in Chapters 7 and 8.

6.8–1 Bhattacharyya and Chernov Bounds

Let us consider a memoryless channel with input alphabet $\mathscr{X}$ and output alphabet $\mathscr{Y}$ which is characterized by the conditional PDF $p(y|x)$. By the memoryless assumption of the channel

$$p(\boldsymbol{y}|\boldsymbol{x}) = \prod_{i=1}^{n} p(y_i|x_i) \tag{6.8–1}$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)$ are input and output sequences of length n. We further assume that from all possible input sequences of length n, a subset of size $M = 2^k$ denoted by $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M$ and called *codewords* is used for transmission. Let us represent by $P_{e|m}$ the error probability when $\boldsymbol{x}_m$ is transmitted and a maximum-likelihood detector is employed. By the union bound and using Equations 4.2–64 to 4.2–67 we can write

$$\begin{aligned} P_{e|m} &= \sum_{\substack{m'=1 \\ m'\neq m}}^{M} \mathrm{P}\left[\boldsymbol{y} \in D_{m'} \,|\boldsymbol{x}_m \text{ sent}\right] \\ &\leq \sum_{\substack{m'=1 \\ m'\neq m}}^{M} \mathrm{P}\left[\boldsymbol{y} \in D_{mm'} \,|\boldsymbol{x}_m \text{ sent}\right] \end{aligned} \tag{6.8–2}$$

where $D_{mm'}$ denotes the decision region for m' in a binary system consisting of $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ and is given by

$$\begin{aligned} D_{mm'} &= \{\boldsymbol{y} : p(\boldsymbol{y}|\boldsymbol{x}_{m'}) > p(\boldsymbol{y}|\boldsymbol{x}_m)\} \\ &= \left\{\boldsymbol{y} : \ln \frac{p(\boldsymbol{y}|\boldsymbol{x}_{m'})}{p(\boldsymbol{y}|\boldsymbol{x}_m)} > 0\right\} \\ &= \{\boldsymbol{y} : Z_{mm'} > 0\} \end{aligned} \tag{6.8–3}$$

in which we have defined

$$Z_{mm'} = \ln \frac{p(\boldsymbol{y}|\boldsymbol{x}_{m'})}{p(\boldsymbol{y}|\boldsymbol{x}_m)} \tag{6.8–4}$$

As in Section 4.2–3, we denote $\mathrm{P}\left[\boldsymbol{y} \in D_{mm'} \,|\boldsymbol{x}_m \text{ sent}\right]$ by $P_{m\to m'}$ and call it *pairwise error probability*, or PEP. It is clear from Equation 6.8–3 that

$$\begin{aligned} P_{m\to m'} &= \mathrm{P}\left[Z_{mm'} > 0 \,|\boldsymbol{x}_m\right] \\ &\leq \mathrm{E}\left[e^{\lambda Z_{mm'}} |\boldsymbol{x}_m\right] \end{aligned} \tag{6.8–5}$$

where in the last step we have used the Chernov bound given by Equation 2.4–4, and the inequality is satisfied for all $\lambda > 0$. Substituting for $Z_{mm'}$ from Equation 6.8–4, we obtain

$$\begin{aligned} P_{m\to m'} &\leq \sum_{\boldsymbol{y}\in\mathscr{Y}^n} e^{\lambda \ln \frac{p(\boldsymbol{y}|\boldsymbol{x}_{m'})}{p(\boldsymbol{y}|\boldsymbol{x}_m)}} p(\boldsymbol{y}|\boldsymbol{x}_m) \\ &= \sum_{\boldsymbol{y}\in\mathscr{Y}^n} p^{\lambda}(\boldsymbol{y}|\boldsymbol{x}_{m'}) p^{1-\lambda}(\boldsymbol{y}|\boldsymbol{x}_m) \qquad \lambda > 0 \end{aligned} \tag{6.8–6}$$

This is the *Chernov bound for the pairwise error probability.* A simpler form of this bound is obtained when we put $\lambda = \frac{1}{2}$. In this case the resulting bound

$$P_{m\to m'} \le \sum_{\boldsymbol{y}\in\mathscr{Y}^n} \sqrt{p(\boldsymbol{y}|\boldsymbol{x}_m)p(\boldsymbol{y}|\boldsymbol{x}_{m'})} \tag{6.8–7}$$

is called the *Bhattacharyya bound.* If the channel is memoryless, the Chernov bound reduces to

$$P_{m\to m'} \le \prod_{i=1}^{n} \left[\sum_{y_i\in\mathscr{Y}} p^{\lambda}(y_i|x_{m'i})p^{1-\lambda}(y_i|x_{mi})\right] \qquad \lambda > 0 \tag{6.8–8}$$

The Bhattacharyya bound for a memoryless channel is given by

$$P_{m\to m'} \le \prod_{i=1}^{n} \sum_{y_i\in\mathscr{Y}} \sqrt{p(y_i|x_{m'i})p(y_i|x_{mi})} \tag{6.8–9}$$

Let use define two functions $\Delta^{(\lambda)}_{x_1\to x_2}$ and Δ_{x_1,x_2}, called *Chernov and Bhatacharyya parameters*, respectively, as

$$\begin{aligned}\Delta^{(\lambda)}_{x_1\to x_2} &= \sum_{y\in\mathscr{Y}} p^{\lambda}(y|x_2)p^{1-\lambda}(y|x_1)\\ \Delta_{x_1,x_2} &= \sum_{y\in\mathscr{Y}} \sqrt{p(y|x_1)p(y|x_2)}\end{aligned} \tag{6.8–10}$$

Note that $\Delta^{(\lambda)}_{x_1\to x_1} = \Delta_{x_1,x_1} = 1$ for all $x_1 \in \mathscr{X}$. Using these definitions, Equations 6.8–8 and 6.8–9 reduce to

$$P_{m\to m'} \le \prod_{i=1}^{n} \Delta^{(\lambda)}_{x_{mi}\to x_{m'i}} \qquad \lambda > 0 \tag{6.8–11}$$

and

$$P_{m\to m'} \le \prod_{i=1}^{n} \Delta_{x_{mi},x_{m'i}} \tag{6.8–12}$$

EXAMPLE 6.8–1. Assume $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ are two binary sequences of length n which differ in d components; d is called the Hamming distance between the two sequences. If a binary symmetric channel with crossover probability p is employed to transmit $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$, we have

$$\begin{aligned}P_{m\to m'} &\le \prod_{i=1}^{n} \Delta_{x_{mi},x_{m'i}}\\ &= \prod_{\substack{i=1\\ x_{mi}\ne x_{m'i}}}^{n} \sqrt{p(1-p)+(1-p)p}\\ &= \left(\sqrt{4p(1-p)}\right)^{d}\end{aligned} \tag{6.8–13}$$

where we have used the fact that if $x_{mi} = x_{m'i}$, then $\Delta_{x_{mi},x_{m'i}} = 1$.

If, instead of the BSC, we use BPSK modulation over an AWGN channel, in which 0 and 1 in each sequence are mapped into $-\sqrt{\mathcal{E}_c}$ and $+\sqrt{\mathcal{E}_c}$ and $\mathcal{E}_c$ denotes energy per component, we will have

$$\begin{aligned}
P_{m\rightarrow m'} &\leq \prod_{i=1}^{n} \Delta_{x_{mi},x_{m'i}} \\
&= \prod_{\substack{i=1\\ x_{mi}\neq x_{m'i}}}^{n} \int_{-\infty}^{\infty} \sqrt{\frac{1}{\pi N_0} e^{-\frac{(y-\sqrt{\mathcal{E}_c})^2}{N_0}} e^{-\frac{(y+\sqrt{\mathcal{E}_c})^2}{N_0}}}\, dy \\
&= \prod_{\substack{i=1\\ x_{mi}\neq x_{m'i}}}^{n} \left(e^{-\frac{\mathcal{E}_c}{N_0}} \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi N_0}} e^{-\frac{y^2}{N_0}}\, dy \right) \\
&= \left(e^{-\frac{\mathcal{E}_c}{N_0}} \right)^{d}
\end{aligned} \tag{6.8–14}$$

In both cases the Bhattacharyya bound is of the form Δ^d, where for the BSC $\Delta = \sqrt{4p(1-p)}$ and for an AWGN channel with BPSK modulation $\Delta = e^{-\frac{\mathcal{E}_c}{N_0}}$. If $p \neq \frac{1}{2}$ and $\mathcal{E}_c > 0$, in both cases $\Delta < 1$ and therefore as d becomes large, the error probability goes to zero.

6.8–2 Random Coding

Let us assume that instead of having two specific codewords $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$, we generate all M codewords according to some PDF $p(x)$ on the input alphabet $\mathscr{X}$. We assume that all codeword components and all codewords are drawn independently according to $p(x)$. Therefore, each codeword $\boldsymbol{x}_m = (x_{m1}, x_{m2}, \ldots, x_{mn})$ is generated according to $\prod_{i=1}^{n} p(x_{mi})$. If we denote the average of the pairwise error probability over the set of randomly generated codes by $\overline{P_{m\rightarrow m'}}$, we have

$$\begin{aligned}
\overline{P_{m\rightarrow m'}} &= \sum_{\boldsymbol{x}_m\in\mathscr{X}^n} \sum_{\boldsymbol{x}_{m'}\in\mathscr{X}^n} P_{m\rightarrow m'} \\
&\leq \sum_{\boldsymbol{x}_m\in\mathscr{X}^n} \sum_{\boldsymbol{x}_{m'}\in\mathscr{X}^n} \prod_{i=1}^{n} \left(p(x_{mi})p(x_{m'i})\Delta^{(\lambda)}_{x_{mi}\rightarrow x_{m'i}} \right) \\
&= \prod_{i=1}^{n} \left(\sum_{x_{mi}\in\mathscr{X}} \sum_{x_{m'i}\in\mathscr{X}} p(x_{mi})p(x_{m'i})\Delta^{(\lambda)}_{x_{mi}\rightarrow x_{m'i}} \right) \\
&= \left(\sum_{x_1\in\mathscr{X}} \sum_{x_2\in\mathscr{X}} p(x_1)p(x_2)\Delta^{(\lambda)}_{x_1\rightarrow x_2} \right)^{n} \qquad \lambda > 0
\end{aligned} \tag{6.8–15}$$

Let us define

$$
\begin{aligned}
R_0(p, \lambda) &= -\log_2 \left[\sum_{x_1 \in \mathscr{X}} \sum_{x_2 \in \mathscr{X}} p(x_1) p(x_2) \Delta^{(\lambda)}_{x_1 \to x_2} \right] \qquad \lambda > 0 \\
&= -\log_2 \left[\mathrm{E}\left[\Delta^{(\lambda)}_{X_1 \to X_2} \right] \right]
\end{aligned}
\tag{6.8–16}
$$

where X_1 and X_2 are independent random variables with joint PDF $p(x_1)p(x_2)$. Using this definition, Equation 6.8–15 can be written as

$$
\overline{P_{m \to m'}} \le 2^{-nR_0(p,\lambda)} \qquad \lambda > 0 \tag{6.8–17}
$$

We define $\overline{P_{e|m}}$ as the average of $P_{e|m}$ over the set of random codes generated using $p(x)$. Using this definition and Equation 6.8–2, we obtain

$$
\begin{aligned}
\overline{P_{e|m}} &\le \sum_{\substack{m'=1 \\ m' \ne m}}^{M} \overline{P_{m \to m'}} \\
&\le \sum_{\substack{m'=1 \\ m' \ne m}}^{M} 2^{-nR_0(p,\lambda)} \\
&= 2^{-n(R_0(p,\lambda) - R_c)} \qquad \lambda > 0
\end{aligned}
\tag{6.8–18}
$$

We have used the relation $M = 2^k = 2^{nR_c}$, where $R_c = \frac{k}{n}$ denotes the rate of the code. Since the right-hand side of the inequality is independent of m, by averaging over m we have

$$
\overline{P_e} \le 2^{-n(R_0(p,\lambda) - R_c)} \qquad \lambda > 0 \tag{6.8–19}
$$

where $\overline{P_e}$ is the average error probability over the ensemble of random codes generated according to $p(x)$. Equation 6.8–19 states that if $R_c \le R_0(p, \lambda)$, for some input PDF $p(x)$ and some $\lambda > 0$, then for n large enough, the average error probability over the ensemble of codes can be made arbitrarily small. This means that among the set of codes generated randomly, there must exist at least one code for which the error probability goes to zero as $n \to \infty$. This is an example of the *random coding* argument first introduced by Shannon in the proof of the channel capacity theorem.

The maximum value of $R_0(p, \lambda)$ over all probability density functions $p(x)$ and all $\lambda > 0$ gives the quantity R_0, known as the *channel cutoff rate*, defined by

$$
\begin{aligned}
R_0 &= \max_{p(x)} \sup_{\lambda > 0} \; R_0(p, \lambda) \\
&= \max_{p(x)} \sup_{\lambda > 0} \; -\log_2 \left[\mathrm{E}\left[\Delta^{(\lambda)}_{X_1 \to X_2} \right] \right]
\end{aligned}
\tag{6.8–20}
$$

Clearly if either $\mathscr{X}$ or $\mathscr{Y}$ or both are continuous,. the corresponding sums in the development of R_0 are substituted with appropriate integrals.

For symmetric channels, the optimal value of λ that maximizes the cutoff rate is $\lambda = \frac{1}{2}$ for which the Chernov bound reduces to the Bhattacharyya bound and

$$\begin{aligned} R_0 &= \max_{p(x)} \; -\log_2 \left[\mathrm{E}\left[\Delta_{X_1, X_2} \right] \right] \\ &= \max_{p(x)} \; -\log_2 \left[\sum_{y \in \mathscr{Y}} \left(\sum_{x \in \mathscr{X}} p(x) \sqrt{p(y|x)} \right)^2 \right] \end{aligned} \tag{6.8–21}$$

In addition to these channels, the PDF maximizing $R_0(p, \lambda)$ is a uniform PDF; i.e., if $Q = |\mathscr{X}|$, we have $p(x) = \frac{1}{Q}$ for all $x \in \mathscr{X}$. In this case we have

$$\begin{aligned} R_0 &= -\log_2 \left[\frac{1}{Q^2} \sum_{y \in \mathscr{Y}} \left(\sum_{x \in \mathscr{X}} \sqrt{p(y|x)} \right)^2 \right] \\ &= 2 \log_2 Q - \log_2 \left[\sum_{y \in \mathscr{Y}} \left(\sum_{x \in \mathscr{X}} \sqrt{p(y|x)} \right)^2 \right] \end{aligned} \tag{6.8–22}$$

Using the inequality

$$\left(\sum_{x \in \mathscr{X}} \sqrt{p(y|x)} \right)^2 \geq \sum_{x \in \mathscr{X}} p(y|x) \tag{6.8–23}$$

and summing over all y, we obtain

$$\begin{aligned} \sum_{y \in \mathscr{Y}} \left(\sum_{x \in \mathscr{X}} \sqrt{p(y|x)} \right)^2 &\geq \sum_{x \in \mathscr{X}} \sum_{y \in \mathscr{Y}} p(y|x) \\ &= Q \end{aligned} \tag{6.8–24}$$

Employing this result in Equation 6.8–22 yields

$$\begin{aligned} R_0 &= 2 \log_2 Q - \log_2 \left[\sum_{y \in \mathscr{Y}} \left(\sum_{x \in \mathscr{X}} \sqrt{p(y|x)} \right)^2 \right] \\ &\leq \log_2 Q \end{aligned} \tag{6.8–25}$$

as expected.

For a symmetric binary-input channel, these relations can be further reduced. In this case

$$\Delta_{x_1, x_2} = \begin{cases} \Delta & x_1 \neq x_2 \\ 1 & x_1 = x_2 \end{cases} \tag{6.8–26}$$

where Δ is the *Bhattacharyya parameter for the binary input channel.* In this case $Q = 2$ and we obtain

$$\begin{aligned} R_0 &= -\log_2 \frac{1 + \Delta}{2} \\ &= 1 - \log_2 (1 + \Delta) \end{aligned} \tag{6.8–27}$$

Since reliable communication is possible at all rates lower than the cutoff rate, we conclude that $R_0 \leq C$. In fact, we can interpret the cutoff rate as the supremum of the rates at which a bound on the average error probability of the form $2^{-n(R_0-R_c)}$ is possible. The simplicity of the exponent in this bound is particularly attractive in comparison with the the general form of the bound on error probability given by $2^{-nE(R_c)}$, where $E(R_c)$ denotes the channel reliability function. Note that $R_0 - R_c$ is positive for all rates less than R_0, but $E(R_c)$ is positive for all rates less than capacity. We will see in Chapter 8 that sequential decoding of convolutional codes is practical at rates lower than R_0. Therefore, we can also interpret R_0 as the supremum of the rates at which sequential decoding is practical.

EXAMPLE 6.8-2. For a BSC, with crossover probability p we have $\mathscr{X} = \mathscr{Y} = \{0, 1\}$. Using the symmetry of the channel, the optimal λ is $\frac{1}{2}$ and the optimal input distribution is a uniform distribution. Therefore,

$$\begin{aligned} R_0 &= 2\log_2 2 - \log_2 \sum_{y=0,1} \left(\sum_{x=0,1} \sqrt{p(y|x)} \right)^2 \\ &= 2\log_2 2 - \log_2 \left[\left(\sqrt{1-p} + \sqrt{p}\right)^2 + \left(\sqrt{p} + \sqrt{1-p}\right)^2 \right] \\ &= 2\log_2 2 - \log_2 \left(2 + 4\sqrt{p(1-p)}\right) \\ &= \log_2 \frac{2}{1 + \sqrt{4p(1-p)}} \end{aligned} \tag{6.8–28}$$

We could also use the fact that $\Delta = \sqrt{4p(1-p)}$ and use Equation 6.8–27 to obtain

$$R_0 = 1 - \log_2(1 + \Delta) = 1 - \log_2 \left(1 + \sqrt{4p(1-p)}\right) \tag{6.8–29}$$

A plot of R_0 versus p is shown in Figure 6.8–1. The capacity of this channel $C = 1 - H_b(p)$ is also shown on the same plot. It is observed that $C \geq R_0$, for all p.

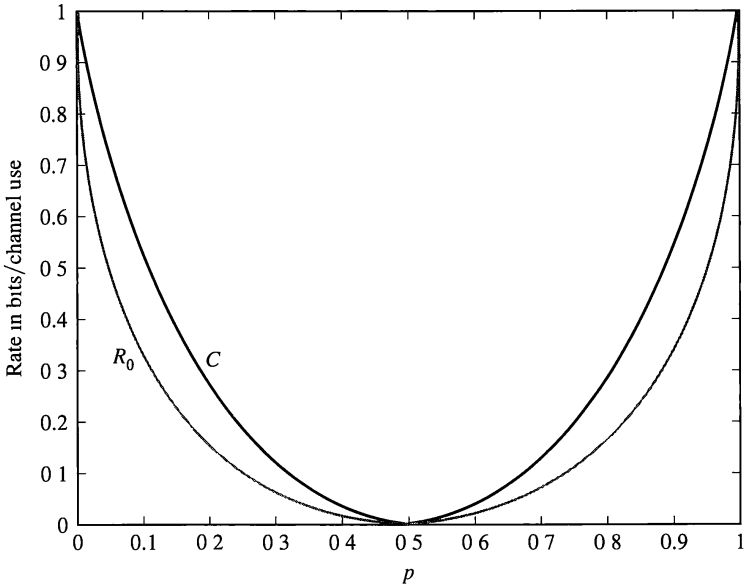


FIGURE 6.8–1
Cutoff rate and channel capacity plots for a binary symmetric channel.

If the BSC channel is obtained by binary quantization of the output of an AWGN channel using BPSK modulation, we have

$$p = Q\left(\sqrt{\frac{2\mathcal{E}_c}{N_0}}\right) \tag{6.8–30}$$

where $\mathcal{E}_c$ denotes energy per component of $\boldsymbol{x}$. Note that with this notation the total energy in $\boldsymbol{x}$ is $\mathcal{E} = n\mathcal{E}_c$; and since each $\boldsymbol{x}$ carries $k = \log_2 M$ bits of information, we have $\mathcal{E}_b = \frac{\mathcal{E}}{k} = \frac{n}{k}\mathcal{E}_c$, or $\mathcal{E}_c = R_c\mathcal{E}_b$, where $R_c = \frac{k}{n}$ is the rate of the code. If the rate of the code tends to R_0, we will have

$$p = Q\left(\sqrt{R_0\gamma_b}\right) \tag{6.8–31}$$

where $\gamma_b = \mathcal{E}_b/N_0$. From the pair of relations

$$\begin{aligned} p &= Q\left(\sqrt{R_0\gamma_b}\right) \\ R_0 &= \log_2 \frac{2}{1+\sqrt{4p(1-p)}} \end{aligned} \tag{6.8–32}$$

we can plot R_0 as a function of γ_b. Similarly, from the pair of relations

$$\begin{aligned} p &= Q\left(\sqrt{R_0\gamma_b}\right) \\ C &= 1 - H_b(p) \end{aligned} \tag{6.8–33}$$

we can plot C as a function of γ_b. These plots that compare R_0 and C as functions of γ_b are shown in Figure 6.8–2. From this figure it is seen that there exists a gap of roughly 2–2.5 dB between R_0 and C.

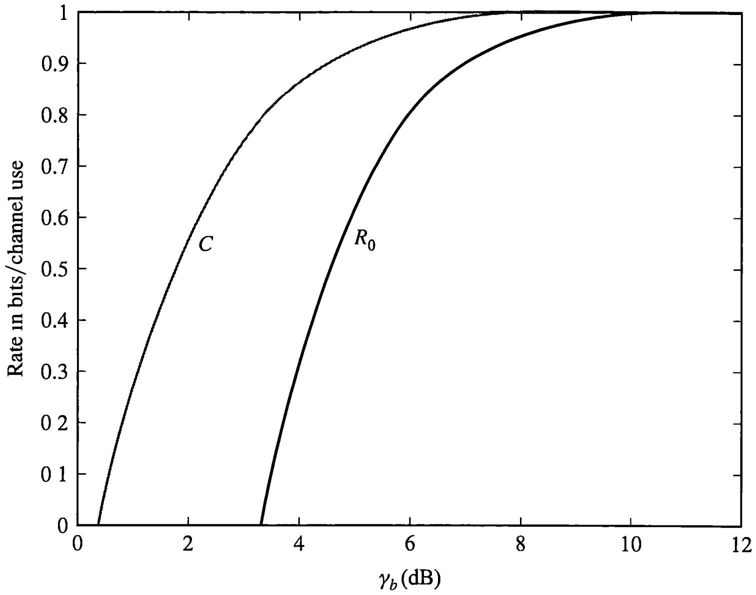


FIGURE 6.8–2
Capacity and cutoff rate for an output quantized BPSK scheme.

EXAMPLE 6.8–3. For an AWGN channel with BPSK modulation we have $\mathscr{X} = \{\pm\sqrt{\mathcal{E}_c}\}$. The output alphabet $\mathscr{Y}$ in this case is the set of real numbers $\mathbb{R}$. We have

$$\begin{aligned}\int_{-\infty}^{\infty}\left(\sum_{x\in\{-\sqrt{\mathcal{E}_c},\sqrt{\mathcal{E}_c}\}}\sqrt{p(y|x)}\right)^2 dy &= \int_{-\infty}^{\infty}\left(\sqrt{\frac{1}{\sqrt{\pi N_0}}e^{-\frac{(y+\sqrt{\mathcal{E}_c})^2}{N_0}}}+\sqrt{\frac{1}{\sqrt{\pi N_0}}e^{-\frac{(y-\sqrt{\mathcal{E}_c})^2}{N_0}}}\right)^2 dy\\ &= 2+2\frac{1}{\sqrt{\pi N_0}}\int_{-\infty}^{\infty}e^{-\frac{y^2+\mathcal{E}_c}{N_0}}\,dy\\ &= 2+2e^{-\frac{\mathcal{E}_c}{N_0}}\end{aligned} \tag{6.8–34}$$

Finally, using Equation 6.8–22, we have

$$\begin{aligned}R_0 &= 2\log_2 2-\log_2\left(2+2e^{-\frac{\mathcal{E}_c}{N_0}}\right)\\ &= \log_2\frac{2}{1+e^{-\frac{\mathcal{E}_c}{N_0}}}\\ &= \log_2\frac{2}{1+e^{-R_c\frac{\mathcal{E}_b}{N_0}}}\end{aligned} \tag{6.8–35}$$

Here $\Delta = e^{-\mathcal{E}_c/N_0}$ and using Equation 6.8–27 will result in the same expression for R_0. A plot of R_0, as well as capacity for this channel which is given by Equation 6.5–31, is shown in Figure 6.8–3.

In Figure 6.8–4 plots of R_0 and C for BPSK with continuous output (soft decision) and BPSK with binary quantized output (hard decision) are compared.

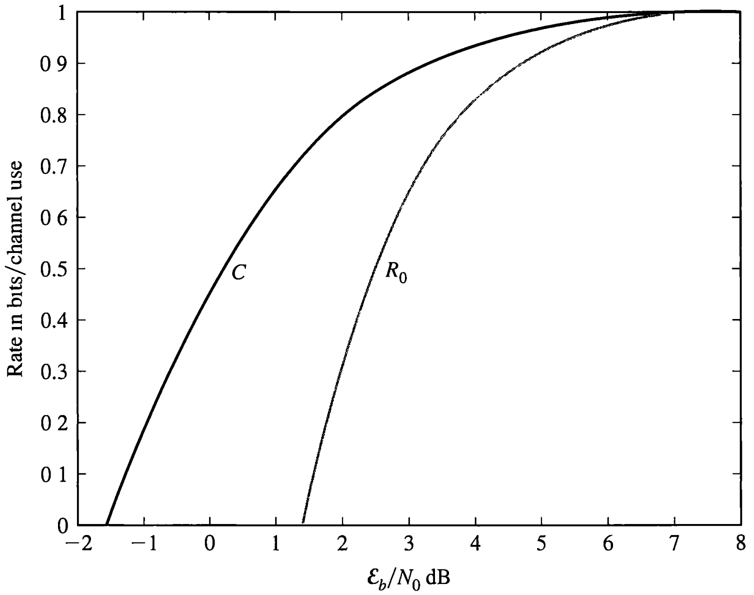


FIGURE 6.8–3
Cutoff rate and channel capacity plots for an AWGN channel with BPSK modulation.

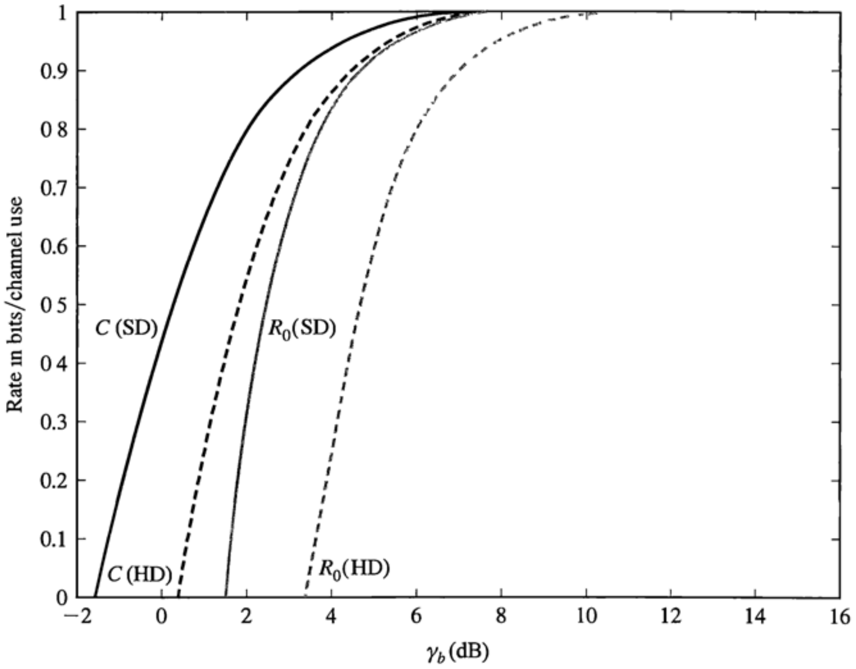


FIGURE 6.8–4
Capacity and cutoff rate for a hard and soft decision decoding of a BPSK scheme.

Comparing the R_0's for hard and soft decisions, we observe that soft decision has an advantage of roughly 2 dB over hard decision. If we compare capacities, we observe a similar 2-dB advantage for soft decision. Comparing R_0 and C, we observe that in both soft and hard decisions, capacity has an advantage of roughly 2–2.5 dB over R_0. This gap is larger at lower SNRs and decreases to 2 dB at higher SNRs.

6.9
BIBLIOGRAPHICAL NOTES AND REFERENCES

Information theory, the mathematical theory of communication, was founded by Shannon (1948, 1959). Source coding has been an area of intense research activity since the publication of Shannon's classic papers in 1948 and the paper by Huffman (1952). Over the years, major advances have been made in the development of highly efficient source data compression algorithms. Of particular significance is the research on universal source coding and universal quantization published by Ziv (1985), Ziv and Lempel (1977, 1978), Davisson (1973), Gray (1975), and Davisson et al. (1981).

Treatments of rate distortion theory are found in the books by Gallager (1968), Berger (1971), Viterbi and Omura (1979), Blahut (1987), and Gray (1990). For practical applications of rate distortion theory to image and video compression, the reader is referred to the *IEEE Signal Processing Magazine*, November 1998, and to the book by Gibson et al. (1998). The paper by Berger and Gibson (1998) on lossy source coding provides an overview of the major developments on this topic over the past 50 years.

Over the past decade, we have also seen a number of important developments in vector quantization. A comprehensive treatment of vector quantization and signal compression is provided in the book of Gersho and Gray (1992). The survey paper by Gray and Neuhoff (1998) describes the numerous advances that have been made on the topic of quantization over the past 50 years and includes a list of over 500 references.

Pioneering work on channel characterization in terms of channel capacity and random coding was done by Shannon (1948a, b; 1949). Additional contributions were subsequently made by Gilbert (1952), Elias (1955), Gallager (1965), Wyner (1965), Shannon et al. (1967), Forney (1968), and Viterbi (1969). All these early publications are contained in the IEEE Press book entitled *Key Papers in the Development of Information Theory*, edited by Slepian (1974). The paper by Verdú (1998) in the 50th Anniversary Commemorative Issue of the *Transactions on Information Theory* gives a historical perspective of the numerous advances in information theory over the past 50 years.

The use of the cutoff rate parameter as a design criterion was proposed and developed by Wozencraft and Kennedy (1966) and by Wozencraft and Jacobs (1965). It was used by Jordan (1966) in the design of coded waveforms for M-ary orthogonal signals with coherent and noncoherent detection. Following these pioneering works, the cutoff rate has been widely used as a design criterion for coded signals in a variety of different channel conditions.

For comprehensive study of the ideas introduced in this chapter, the reader is referred to standard texts on information theory including Gallager (1968) and Cover and Thomas (2006).

PROBLEMS

6.1 Prove that $\ln u \leq u - 1$ and also demonstrate the validity of this inequality by plotting $\ln u$ and $u - 1$ on the same graph.

6.2 X and Y are two discrete random variables with probabilities

$$P(X = x, Y = y) \equiv P(x, y)$$

Show that $I(X; Y) \geq 0$, with equality if and only if X and Y are statistically independent. *Hint*: Use the inequality $\ln u \leq u - 1$, for $0 < u < 1$, to show that $-I(X; Y) \leq 0$.

6.3 The output of a DMS consists of the possible letters $x_1, x_2, \ldots, x_n$, which occur with probabilities $p_1, p_2, \ldots, p_n$, respectively. Prove that the entropy $H(X)$ of the source is at most $\log n$. Find the probability density function for which $H(X) = \log n$.

6.4 Let X be a geometrically distributed random variable, i.e.,

$$P(X = k) = p(1 - p)^{k-1}, \qquad k = 1, 2, 3, \ldots$$

1. Find the entropy of X.
2. Given that $X > K$, where K is a positive integer, what is the entropy of X?

6.5 Two binary random variables X and Y are distributed according to the joint distributions $P(X = Y = 0) = P(X = 0, Y = 1) = P(X = Y = 1) = \frac{1}{3}$. Compute $H(X)$, $H(Y)$, $H(X|Y)$, $H(Y|X)$, and $H(X, Y)$.

6.6 Let X and Y denote two jointly distributed, discrete-valued random variables.

1. Show that

$$H(X) = -\sum_{x,y} P(x, y) \log P(x)$$

and

$$H(Y) = -\sum_{x,y} P(x, y) \log P(v)$$

2. Use the above result to show that

$$H(X, Y) \leq H(X) + H(Y)$$

When does equality hold?

3. Show that

$$H(X|Y) \leq H(X)$$

with equality if and only if X and Y are independent.

6.7 Let $Y = g(X)$, where g denotes a deterministic function. Show that, in general, $H(Y) \leq H(X)$. When does equality hold?

6.8 Show that, for statistically independent events,

$$H(X_1 X_2 \cdots X_n) = \sum_{i=1}^{n} H(X_i)$$

6.9 Show that

$$I(X_3; X_2|X_1) = H(X_3|X_1) - H(X_3|X_1 X_2)$$

and that

$$H(X_3|X_1) \geq H(X_3|X_1 X_2)$$

6.10 Let X be a random variable with PDF $p_X(x)$, and let $Y = aX + b$ be a linear transformation of X, where a and b are two constants. Determine the differential entropy $H(Y)$ in terms of $H(X)$.

6.11 The outputs x_1, x_2, and x_3 of a DMS with corresponding probabilities $p_1 = 0.45$, $p_2 = 0.35$, and $p_3 = 0.20$ are transformed by the linear transformation $Y = aX + b$, where a and b are constants. Determine the entropy $H(Y)$ and comment on what effect the transformation has had on the entropy of X.

6.12 A Markov process is a process with one-step memory, i.e., a process such that

$$p(x_n|x_{n-1}, x_{n-2}, x_{n-3}, \ldots) = p(x_n|x_{n-1})$$

for all n. Show that, for a stationary Markov process, the entropy rate is given by

$$H(X_n|X_{n-1})$$

6.13 A first-order Markov source is characterized by the state probabilities $P(x_i), i = 1, 2, \ldots, L$, and the transition probabilities $P(x_k|x_i),\ k = 1, 2, \ldots, L$, and $k \neq i$. The entropy of the Markov source is

$$H(X) = \sum_{k=1}^{L} P(x_k)H(X|x_k)$$

where $H(X|x_k)$ is the entropy conditioned on the source being in state x_k. Determine the entropy of the binary, first-order Markov source shown in Figure P6.13, which has the transition probabilities $P(x_2|x_1) = 0.2$ and $P(x_1|x_2) = 0.3$. Note that the conditional entropies $H(X|x_1)$ and $H(X|x_2)$ are given by the binary entropy functions $H_b(P(x_2|x_1))$ and $H_b(P(x_1|x_2))$, respectively. How does the entropy of the Markov source compare with the entropy of a binary DMS with the same output letter probabilities $P(x_1)$ and $P(x_2)$?

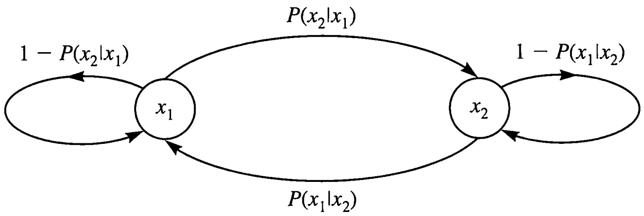


FIGURE P6.13

6.14 Show that, for a DMC, the average mutual information between a sequence $X_1, X_2, \ldots, X_n$ of channel inputs and the corresponding channel outputs satisfies the condition

$$I(X_1X_2\cdots X_n; Y_1Y_2\cdots Y_n) \leq \sum_{i=1}^{n} I(X_i; Y_i)$$

with equality if and only if the set of input symbols is statistically independent.

6.15 Determine the differential entropy $H(X)$ of the uniformly distributed random variable X with PDF

$$p(x) = \begin{cases} a^{-1} & 0 \leq x \leq a \\ 0 & \text{otherwise} \end{cases}$$

for the following three cases:

1. $a = 1$
2. $a = 4$
3. $a = \frac{1}{4}$

Observe from these results that $H(X)$ is not an absolute measure, but only a relative measure of randomness.

6.16 A DMS has an alphabet of five letters $x_i,\ i = 1, 2, \ldots, 5$, each occurring with probability $\frac{1}{5}$. Evaluate the efficiency of a fixed-length binary code in which

1. Each letter is encoded separately into a binary sequence.
2. Two letters at a time are encoded into a binary sequence.
3. Three letters at a time are encoded into a binary sequence.

6.17 Determine whether there exists a binary code with codeword lengths $(n_1, n_2, n_3, n_4) = (1, 2, 2, 3)$ that satisfy the prefix condition.

6.18 Consider a binary block code with 2^n codewords of the same length n. Show that the Kraft inequality is satisfied for such a code.

6.19 A DMS has an alphabet of eight letters x_i, $i = 1, 2, \ldots, 8$, with probabilities 0.25, 0.20, 0.15, 0.12, 0.10, 0.08, 0.05, and 0.05.

1. Use the Huffman encoding procedure to determine a binary code for the source output.
2. Determine the average number $\overline{R}$ of binary digits per source letter.
3. Determine the entropy of the source and compare it with $\overline{R}$.

6.20 A discrete memoryless source produces outputs $\{a_1, a_2, a_3, a_4, a_5, a_6\}$. The corresponding output probabilities are $0.7, 0.1, 0.1, 0.05, 0.04$, and 0.01.

1. Design a binary Huffman code for the source. Find the average codeword length. Compare it to the minimum possible average codeword length.
2. Is it possible to transmit this source reliably at a rate of 1.5 bits per source symbol? Why?
3. Is it possible to transmit the source at a rate of 1.5 bits per source symbol employing the Huffman code designed in part 1?

6.21 A discrete memoryless source is described by the alphabet $\mathcal{X} = \{x_1, x_2, \ldots, x_8\}$, and the corresponding probability vector $\boldsymbol{p} = \{0.2, 0.12, 0.06, 0.15, 0.07, 0.1, 0.13, 0.17\}$. Design a Huffman code for this source; find $\bar{L}$, the average codeword length for the Huffman code; and determine the efficiency of the code defined as

$$\eta = \frac{H(X)}{\bar{L}}$$

6.22 The optimum four-level nonuniform quantizer for a Gaussian-distributed signal amplitude results in the four levels a_1, a_2, a_3, and a_4, with corresponding probabilities of occurrence $p_1 = p_2 = 0.3365$ and $p_3 = p_4 = 0.1635$.

1. Design a Huffman code that encodes a single level at a time, and determine the average bit rate.
2. Design a Huffman code that encodes two output levels at a time, and determine the average bit rate.
3. What is the minimum rate obtained by encoding J output levels at a time as $J \to \infty$?

6.23 A discrete memoryless source has an alphabet of size 7, $\mathscr{X} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$, with corresponding probabilities $\{0.02, 0.11, 0.07, 0.21, 0.15, 0.19, 0.25\}$.

1. Determine the entropy of this source.
2. Design a Huffman code for this source, and find the average codeword length of the Huffman code.
3. A new source $\mathscr{Y} = \{y_1, y_2, y_3\}$ is obtained by grouping the outputs of the source $\mathscr{X}$ as

$$y_1 = \{x_1, x_2, x_5\}$$
$$y_2 = \{x_3, x_7\}$$
$$y_3 = \{x_4, x_6\}$$

   Determine the entropy of $\mathscr{Y}$.
4. Which source is more predictable, $\mathscr{X}$ or $\mathscr{Y}$? Why?

6.24 An iid source $\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots$ has the pdf

$$f(x) = \begin{cases} e^{-x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

This source is quantized using the following scheme:

$$\hat{X} = \begin{cases} 0.5 & 0 \leq X < 1 \\ 1.5 & 1 \leq X < 2 \\ 2.5 & 2 \leq X < 3 \\ 3.5 & 3 \leq X < 4 \\ 6 & \text{otherwise} \end{cases}$$

1. Design a Huffman code for the quantized source $\hat{X}$.
2. What is the entropy of the quantized source $\hat{X}$?
3. If the efficiency of the Huffman code is defined as the ratio of the entropy to the average codeword length of the Huffman code, determine the efficiency of the Huffman code designed in part 1.
4. Now let $\tilde{X} = i + 0.5, i \leq X < i + 1$, for $i = 0, 1, 2, \ldots$. Which random variable has a higher entropy, $\hat{X}$ or $\tilde{X}$? (There is no need to compute entropy of $\tilde{X}$, just give your intuitive reasoning.)

6.25 A stationary source generates outputs at a rate of 10,000 samples. The samples are independent and are uniformly distributed on the interval $[-4, 4]$. Throughout this problem the distortion measure is assumed to be squared-error distortion.

1. If perfect (distortion-free) reconstruction of the source at the destination is required, what is the required transmission rate from the source to the destination?
2. If the transmission rate from the source to the destination is zero, what is the minimum achievable distortion?
3. If a five-level uniform quantizer is designed for this source and the quantizer output is entropy-coded using a Huffman code designed for single-source outputs, what is the resulting transmission rate and distortion?
4. In part 3 if the Huffman code is designed for very large blocks of source outputs rather than single source outputs, what is the resulting transmission rate and distortion?

6.26 A memoryless source has the alphabet $A = \{-5, -3, -1, 0, 1, 3, 5\}$, with corresponding probabilities $\{0.05, 0.1, 0.1, 0.15, 0.05, 0.25, 0.3\}$.

1. Find the entropy of the source.
2. Assuming that the source is quantized according to the quantization rule

$$\begin{cases} q(-5) = q(-3) = -4 \\ q(-1) = q(0) = q(1) = 0 \\ q(3) = q(5) = 4 \end{cases}$$

find the entropy of the quantized source.

6.27 Design a *ternary* Huffman code, using 0, 1, and 2 as letters, for a source with output alphabet probabilities given by $\{0.05, 0.1, 0.15, 0.17, 0.18, 0.22, 0.13\}$. What is the resulting average codeword length? Compare the average codeword length with the entropy of the source. (In what base would you compute the logarithms in the expression for the entropy for a meaningful comparison?)

6.28 Two discrete memoryless information sources X and Y each have an alphabet with six symbols, $\mathscr{X} = \mathscr{Y} = \{1, 2, 3, 4, 5, 6\}$. The probabilities of the letters for X are $1/2, 1/4, 1/8, 1/16, 1/32$, and $1/32$. The source Y has a uniform distribution.

1. Which source is less predictable and why?
2. Design Huffman codes for each source. Which Huffman code is more efficient? (Efficiency of a Huffman code is defined as the ratio of the source entropy to the average codeword length.)
3. If Huffman codes were designed for the second extension of these sources (i.e., two letters at a time), for which source would you expect a performance improvement compared to the single-letter Huffman code and why?
4. Now assume the two sources are independent and a new source Z is defined to be the sum of the two sources, i.e., $Z = X + Y$. Determine the entropy of this source, and verify that $H(Z) < H(X) + H(Y)$.
5. How do you justify the fact that $H(Z) < H(X) + H(Y)$? Under what circumstances can you have $H(Z) = H(X) + H(Y)$? Is there a case where you can have $H(Z) > H(X) + H(Y)$? Why?

6.29 A function $g(x)$ is *convex* on (a, b) if for any $x_1, x_2 \in (a, b)$ and any $0 \leq \lambda \leq 1$

$$g(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda g(x_1) + (1 - \lambda)g(x_2)$$

The function $g(x)$ is convex if its second derivative is nonnegative in the given interval. A function $g(x)$ is called *concave* if $-g(x)$ is convex.

1. Show that $H_b(p)$, the binary entropy function, is concave on (0, 1).
2. Show that $Q(x)$ is convex on $(0, \infty)$.
3. Show that if X is a binary-valued random variable with range in (a, b) and $g(X)$ is convex on (a, b), then

$$g\left(\mathrm{E}\left[X\right]\right) \leq \mathrm{E}\left[g(X)\right]$$

4. Extend the result of part 3 to any random variable X with range in (a, b). This result is known as *Jensen's inequality*.
5. Use Jensen's inequality to prove that if X is a positive-valued random variable, then

$$\mathrm{E}\left[Q(X)\right] \geq Q\left(\mathrm{E}\left[X\right]\right)$$

6.30 Find the Lempel Ziv source code for the binary source sequence

000100100000011000010000000100000010100001000000110100000001100

Recover the original sequence back from the Lempel Ziv source code. *Hint*: You require two passes of the binary sequence to decide on the size of the dictionary.

6.31 A continuous-valued, discrete-time, iid (independent and identically distributed) information source $\ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots$ has the probability density function (PDF) given by

$$f(x) = \begin{cases} \frac{1}{2}e^{-\frac{x}{2}} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

This source is quantized to source $\hat{X}$ using the following quantization rule:

$$\hat{X} = \begin{cases} 0.5 & 0 \leq X < 1 \\ 1.5 & 1 \leq X < 2 \\ 2.5 & 2 \leq X < 3 \\ 6 & \text{otherwise} \end{cases}$$

1. What is the minimum required rate for lossless transmission of the nonquantized source X?
2. What is the minimum required rate for lossless transmission of the quantized source $\hat{X}$?
3. Let $\tilde{X}$ be another quantization of X given by $\tilde{X} = i + 0.25$ if $i \leq X < i + 1$ for $i = 0, 1, 2, \ldots$. Which random variable has a higher entropy, $\hat{X}$ or $\tilde{X}$? (There is no need to compute entropy of $\tilde{X}$, just give your intuitive reasoning.)
4. Let us define a new quantization rule as $Y = \hat{X} + \tilde{X}$. Which of the three relations given below are true (if any)?
   (a) $H(Y) = H(\hat{X}) + H(\tilde{X})$
   (b) $H(Y) = H(\hat{X})$
   (c) $H(Y) = H(\tilde{X})$
   Give your intuitive reason in one short paragraph; no computation is required.

6.32 Find the differential entropy of the continuous random variable X in the following cases:

1. X is an exponential random variable with parameter $\lambda > 0$, i.e.,

$$p(x) = \begin{cases} \frac{1}{\lambda} e^{-x/\lambda} & x > 0 \\ 0 & \text{otherwise} \end{cases}$$

2. X is a Laplacian random variable with parameter $\lambda > 0$, i.e.,

$$p(x) = \frac{1}{2\lambda} e^{-|x|/\lambda}$$

3. X is a triangular random variable with parameter $\lambda > 0$, i.e.,

$$p(x) = \begin{cases} (x + \lambda)/\lambda^2 & -\lambda \leq x \leq 0 \\ (-x + \lambda)/\lambda^2 & 0 < x \leq \lambda \\ 0 & \text{otherwise} \end{cases}$$

6.33 It can be shown that the rate distortion function for a Laplacian source $p(x) = (2\lambda)^{-1} e^{-|x|/\lambda}$ with an absolute value of error distortion measure $d(x, \hat{x}) = |x - \hat{x}|$ is given by

$$R(D) = \begin{cases} \log(\lambda/D) & 0 \leq D \leq \lambda \\ 0 & D > \lambda \end{cases}$$

(see Berger, 1971).

1. How many bits per sample are required to represent the outputs of this source with an average distortion not exceeding $\frac{1}{2}\lambda$?
2. Plot $R(D)$ for three different values of λ, and discuss the effect of changes in λ on these plots.

6.34 Three information sources X, Y, and Z are considered.

1. X is a binary discrete memoryless source with $p(X = 0) = 0.4$. This source is to be reproduced at the receiving end with an error probability not exceeding 0.1.
2. Y is a memoryless Gaussian source with mean 0 and variance 4. This source is to be reproduced with a squared-error distortion not exceeding 1.5.
3. Z is a memoryless source and has a distribution given by

$$f_Z(z) = \begin{cases} 1/5 & -2 \le z \le 0 \\ 3/10 & 0 < z \le 2 \\ 0 & \text{otherwise} \end{cases}$$

This source is quantized using a *uniform* quantizer with eight quantization levels to get the quantized source $\hat{Z}$. The quantized source is required to be transmitted with no errors.

In each of the three cases, determine the *absolute minimum rate required* per source symbol (i.e., you can use systems of arbitrary complexity).

6.35 It can be shown that if X is a zero-mean continuous random variable with variance σ^2, its rate distortion function, subject to squared-error distortion measure, satisfies the lower and upper bounds given by the inequalities

$$H(X) - \frac{1}{2}\log(2\pi e D) \le R(D) \le \frac{1}{2}\log\frac{\sigma^2}{2}$$

where $H(X)$ denotes the differential entropy of the random variable X (see Cover and Thomas, 2006).

1. Show that, for a Gaussian random variable, the lower and upper bounds coincide.
2. Plot the lower and upper bounds for a Laplacian source with $\sigma = 1$.
3. Plot the lower and upper bounds for a triangular source with $\sigma = 1$.

6.36 A DMS has an alphabet of eight letters x_i, $i = 1, 2, \ldots, 8$, with probabilities given in Problem 6.19. Use the Huffman encoding procedure to determine a ternary code (using symbols 0, 1, and 2) for encoding the source output. (*Hint*: Add a symbol x_9 with probability $p_9 = 0$, and group three symbols at a time.)

6.37 Show that the entropy of an n-dimensional Gaussian vector $\boldsymbol{X} = (x_1 x_2 \cdots x_n)$ with zero mean and covariance matrix $\boldsymbol{C}$ is

$$H(\boldsymbol{X}) = \frac{1}{2}\log(2\pi e)^n |\boldsymbol{C}|$$

6.38 Evaluate the rate distortion function for an M-ary symmetric source under Hamming distortion (probability of error) given as

$$R(D) = \log M + D\log D + (1 - D)\log\frac{1 - D}{M - 1}$$

for $M = 2, 4, 8$, and 16.

6.39 Consider the use of the weighted mean square error (MSE) distortion measure defined as

$$d_w(\boldsymbol{X}, \tilde{\boldsymbol{X}}) = (\boldsymbol{X} - \tilde{\boldsymbol{X}})^t \boldsymbol{W}(\boldsymbol{X} - \tilde{\boldsymbol{X}})$$

where $\boldsymbol{W}$ is a symmetric, positive-definitive weighting matrix. By factorizing $\boldsymbol{W}$ as $\boldsymbol{W} = \boldsymbol{P}'\boldsymbol{P}$, show that $d_w(\boldsymbol{X}, \tilde{\boldsymbol{X}})$ is equivalent to an unweighted MSE distortion measure $d_2(\boldsymbol{X}', \tilde{\boldsymbol{X}}')$ involving transformed vectors $\boldsymbol{X}'$ and $\tilde{\boldsymbol{X}}'$.

6.40 A discrete memoryless source produces outputs $\{a_1, a_2, a_3, a_4, a_5\}$. The corresponding output probabilities are 0.8, 0.1, 0.05, 0.04, and 0.01.

1. Design a binary Huffman code for the source. Find the average codeword length. Compare it to the minimum possible average codeword length.
2. Assume that we have a binary symmetric channel with crossover probability $\epsilon = 0.3$. Is it possible to transmit the source reliably over the channel? Why?
3. Is it possible to transmit the source over the channel employing Huffman code designed for single source outputs?

6.41 A discrete-time memoryless Gaussian source with mean 0 and variance σ^2 is to be transmitted over a binary symmetric channel with crossover probability ϵ.

1. What is the minimum value of the distortion attainable at destination? (Distortion is measured in mean squared error.)
2. If the channel is discrete-time memoryless additive Gaussian noise with input power P and noise power σ_n^2, what is the minimum attainable distortion?
3. Now assume that the source has the same basic properties but is not memoryless. Do you expect that the distortion in transmission over the binary symmetric channel to be decreased or increased? Why?

6.42 An additive white Gaussian noise channel has the output $Y = X + N$, where X is the channel input and N is the noise with probability density function

$$p(n) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-n^2/2\sigma_n^2}$$

If X is a white Gaussian input with $E(X) = 0$ and $E(X^2) = \sigma_X^2$, determine

1. The conditional differential entropy $H(X|N)$
2. The mutual information $I(X; Y)$

6.43 For the channel shown in Figure P6.43, find the channel capacity and the input distribution that achieves capacity.

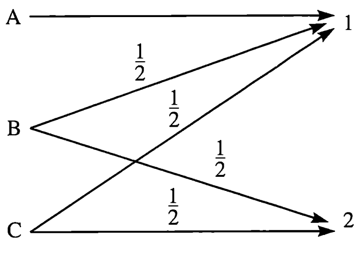


FIGURE P6.43

6.44 A discrete memoryless source produces outputs $\{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8\}$. The corresponding output probabilities are 0.05, 0.07, 0.08, 0.1, 0.1, 0.15, 0.2, and 0.25.

1. Design a binary Huffman code for the source. Find the average codeword length. Compare it to the minimum possible average codeword length.
2. What is the minimum channel capacity required to transmit this source reliably? Can this source be reliably transmitted via a binary symmetric channel?

3. If a discrete memoryless zero-mean Gaussian source with $\sigma^2 = 1$ is to be transmitted via the channel of part 2, what is the minimum attainable mean squared distortion?

6.45 Find the capacity of channels A and B as shown in Figure P6.45. What is the capacity of the cascade channel AB? (*Hint*: Look carefully at the channels, avoid lengthy math.)

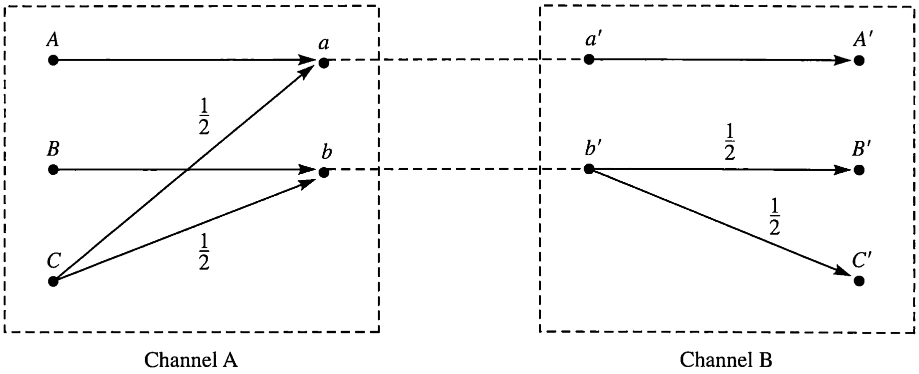


FIGURE P6.45

6.46 Each sample of a Gaussian memoryless source has a variance equal to 4, and the source produces 8000 samples per second. The source is to be transmitted via an additive white Gaussian noise channel with a bandwidth equal to 4000 Hz, and it is desirable to have a distortion per sample not exceeding 1 at the destination (assume squared-error distortion).

1. What is the minimum required signal-to-noise ratio of the channel?
2. If it is further assumed that, on the same channel, a BPSK scheme is employed with hard decision decoding, what will be the minimum required channel signal-to-noise ratio?

Note: the signal-to-noise ratio of the channel is defined by $\frac{P}{N_0 W}$.

6.47 A communication channel is shown in Figure P6.47.

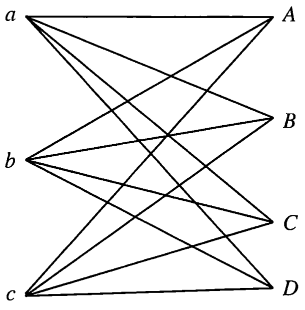


FIGURE P6.47

1. Show that, regardless of the contents of the probability transition matrix of the channel, we have

$$C \leq \log_2 3 \approx 1.585. \text{ bits per transmission}$$

2. Determine one probability transition matrix under which the above upper bound is achieved.

3. Assuming that a Gaussian source with variance $\sigma^2 = 1$ is to be transmitted via the channel in part 2, what is the minimum achievable distortion? (Mean squared distortion is assumed throughout.)

6.48 X is a binary memoryless source with $P(X = 0) = 0.3$. This source is transmitted over a binary symmetric channel with crossover probability $p = 0.1$.

1. Assume that the source is directly connected to the channel; i.e., no coding is employed. What is the error probability at the destination?
2. If coding is allowed, what is the minimum possible error probability in the reconstruction of the source?
3. For what values of p is reliable transmission possible (with coding, of course)?

6.49 Two discrete memoryless information sources S_1 and S_2 each have an alphabet with six symbols, $S_1 = \{x_1, x_2, \ldots, x_6\}$ and $S_2 = \{y_1, y_2, \ldots, y_6\}$. The probabilities of the letters for the first source are 1/2, 1/4, 1/8, 1/16, 1/32, and 1/32. The second source has a uniform distribution.

1. Which source is less predictable and why?
2. Design Huffman codes for each source. Which Huffman code is more efficient? (Efficiency of a Huffman code is defined as the ratio of the source entropy to the average codeword length.)
3. If Huffman codes were designed for the second extension of these sources (i.e., two letters at a time), for which source would you expect a performance improvement compared to the single-letter Huffman code and why?

6.50 Show that the capacity of a binary-input, continuous-output AWGN channel with input-output relation

$$y_i = x_i + n_i$$

where $x_i = \pm A$ and noise components n_i are iid zero-mean Gaussian random variables with variance σ^2 as given by Equations 6.5–31 and 6.5–32.

6.51 A discrete memoryless channel is shown in Figure P6.51.

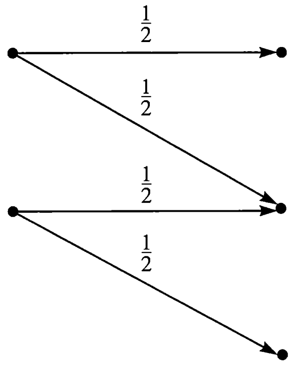


FIGURE P6.51

1. Determine the capacity of this channel.
2. Determine R_0 for this channel.
3. If a discrete-time memoryless Gaussian source with a variance of 4 is to be transmitted by this channel, and for each source output, two uses of channel are allowed, what is the absolute minimum to the achievable squared-error distortion?

6.52 Show that the following two relations are necessary and sufficient conditions for the set of input probabilities $\{P(x_j)\}$ to maximize $I(X;\,Y)$ and, thus, to achieve capacity for a DMC:

$$\begin{aligned} I(x_j;\,Y) &= C \qquad \text{for all } j \text{ with } P(x_j) > 0 \\ I(x_j;\,Y) &\le C \qquad \text{for all } j \text{ with } P(x_j) = 0 \end{aligned}$$

where C is the capacity of the channel, $Q = |\mathscr{Y}|$, and

$$I(x_j;\,Y) = \sum_{i=0}^{Q-1} P(y_i|x_j) \log \frac{P(y_i|x_j)}{P(y_i)}$$

6.53 Figure P6.53 illustrates a M-ary symmetric DMC with transition probabilities $P(y|x) = 1 - p$ when $x = y = k$ for $k = 0, 1, \ldots, M-1$, and $P(y|x) = p/(M-1)$ when $x \neq y$.

1. Show that this channel satisfies the condition given in Problem 6.52 when $P(x_k) = 1/M$.
2. Determine and plot the channel capacity as a function of p.

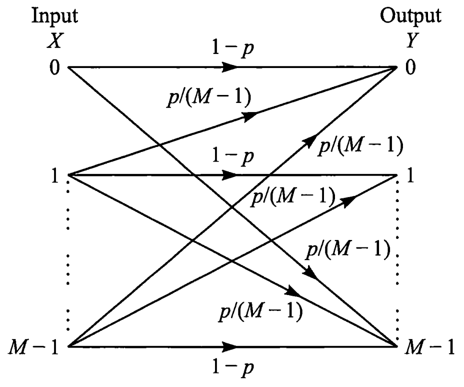
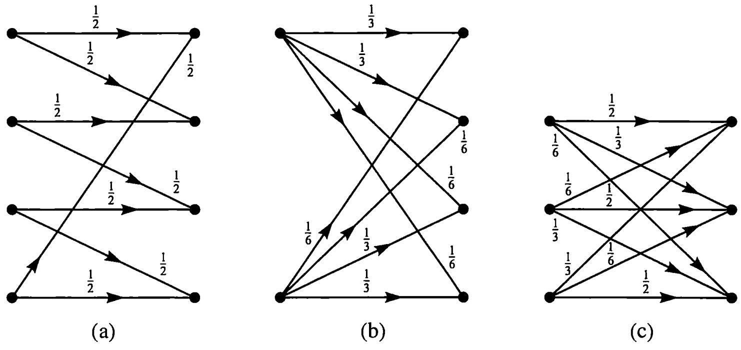


FIGURE P6.53

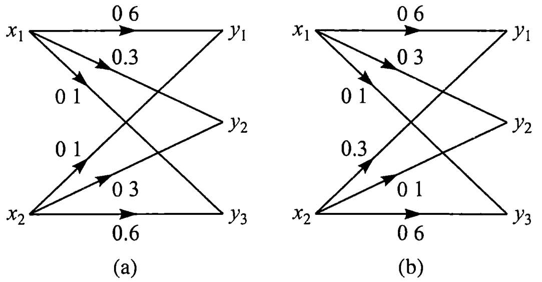
6.54 Determine the capacities of the channels shown in Figure P6.54.



(a) (b) (c)

FIGURE P6.54

6.55 Consider the two channels with the transition probabilities as shown in Figure P6.55. Determine if equally probable input symbols maximize the information rate through the channel.



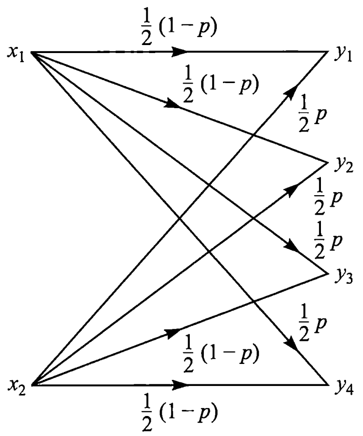
x_1
x_2
y_1
y_2
y_3
0 6
0.3
0 1
0 1
0 3
0.6
(a)
x_1
x_2
y_1
y_2
y_3
0 6
0 3
0 1
0.3
0 1
0 6
(b)

FIGURE P6.55

6.56 A telephone channel has a bandwidth $W = 3000$ Hz and a signal-to-noise power ratio of 400 (26 dB). Suppose we characterize the channel as a band-limited AWGN waveform channel with $P_{\text{av}}/WN_0 = 400$. Determine the capacity of the channel in bits per second.

6.57 Consider the binary-input, quaternary-output DMC shown in Figure P6.57.

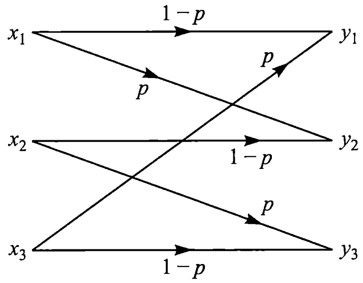
1. Determine the capacity of the channel.
2. Show that this channel is equivalent to a BSC.



x_1
x_2
y_1
y_2
y_3
y_4
$\frac{1}{2}(1-p)$
$\frac{1}{2}(1-p)$
$\frac{1}{2}p$
$\frac{1}{2}p$
$\frac{1}{2}p$
$\frac{1}{2}p$
$\frac{1}{2}(1-p)$
$\frac{1}{2}(1-p)$

FIGURE P6.57

6.58 Determine the capacity for the channel shown in Figure P6.58.



x_1
x_2
x_3
y_1
y_2
y_3
$1-p$
p
p
$1-p$
p
$1-p$

FIGURE P6.58

6.59 Consider a BSC with crossover probability of p. Suppose that R is the number of bits in a source codeword that represents one of 2^R possible levels at the output of a quantizer.

1. Determine the probability that a codeword transmitted over the BSC is received correctly.
2. Determine the probability of having at least one bit error in a codeword transmitted over the BSC.
3. Determine the probability of having n_e or fewer bit errors in a codeword.
4. Evaluate the probabilities in parts 1, 2, and 3 for $R = 5$, $p = 0.1$, and $n_e = 5$.

6.60 Figure P6.60 illustrates a binary erasure channel with transition probabilities $P(0|0) = P(1|1) = 1 - p$ and $P(e|0) = P(e|1) = p$. The probabilities for the input symbols are $P(X = 0) = \alpha$ and $P(X = 1) = 1 - \alpha$.

1. Determine the average mutual information $I(X; Y)$ in bits.
2. Determine the value of α that maximizes $I(X; Y)$, i.e., the channel capacity C in bits per channel use, and plot C as a function of p for the optimum value of α.
3. For the value of α found in part 2, determine the mutual information $I(x; y) = I(0; 0)$, $I(1; 1)$, $I(0; e)$, and $I(1; e)$, where

$$I(x; y) = \log \frac{\mathrm{P}[X = x, Y = y]}{\mathrm{P}[X = x]\,\mathrm{P}[Y = y]}$$

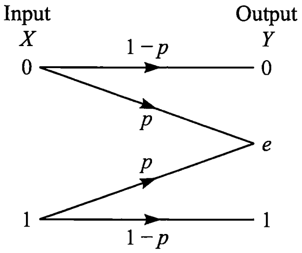


FIGURE P6.60

6.61 A discrete-time zero-mean Gaussian random process has a variance per sample of σ_1^2. This source generates outputs at a rate of 1000 per second. The samples are transmitted over a discrete-time AWGN channel with input power constraint of P and noise variance per sample of σ_2^2. This channel is capable of transmitting 500 symbols per second.

1. If the source is to be transmitted over the channel, you are allowed to employ processing schemes of any degree of complexity, and any delay is acceptable, what is the minimum achievable distortion per sample?
2. If the channel remains the same but you have to use binary antipodal signals at the input and employ hard decision decoding at the output (again no limit on complexity and delay), what is the minimum achievable distortion per sample?
3. Now assume that the source has the same statistics but *is not memoryless*. Comparing with part 1, do you expect the distortion to decrease or increase? Give your answer in a short paragraph.

6.62 A binary memoryless source generates 0 and 1 with probabilities 1/3 and 2/3, respectively. This source is to be transmitted over an AWGN channel using binary PSK modulation.

1. What is the absolute minimum E_b/N_0 required to be able to transmit the source reliably, assuming that hard decision decoding is employed by the channel and for each source output you can use one channel transmission.
2. Under the same conditions as in part 1, find the minimum $\mathcal{E}_b/N_0$ required for reliable transmission of the source if we can transmit at a rate at most equal to the cutoff rate of the channel.
3. Now assume the source is a zero-mean memoryless Gaussian source with variance 1. Answer part 1 if our goal is reproduction of the source with a mean-squared distortion of at most 1/4.

6.63 A discrete memoryless source U is to be transmitted over a memoryless communication channel. For each source output, the channel can be used only once. Determine the minimum theoretical distortion achievable in transmission of the source over the channel in each of the following cases.

1. The source is a binary source with 0 and 1 as its outputs with $p(U = 0) = 0.1$; the channel is a binary symmetric channel with crossover probability $\epsilon = 0.1$; and the distortion measure is the Hamming distortion (probability of error).
2. The channel is as in part 1, but the source is a zero-mean Gaussian source with variance 1. The distortion is the squared-error distortion.
3. The source is as in part 2, and the channel is a discrete-time AWGN channel with input power constraint P and noise variance σ^2.

6.64 Channel C_1 is an additive white Gaussian noise channel with a bandwidth W, average transmitter power P, and noise power spectral density $\frac{1}{2}N_0$. Channel C_2 is an additive Gaussian noise channel with the same bandwidth and power as channel C_1 but with noise power spectral density $\mathcal{S}_n(f)$. It is further assumed that the total noise power for both channels is the same; i.e.,

$$\int_{-W}^{W} \mathcal{S}_n(f)\,df = \int_{-W}^{W} \frac{1}{2}N_0\,df = N_0 W$$

Which channel do you think has a larger capacity? Give an intuitive reasoning.

6.65 A discrete memoryless ternary erasure communication channel is shown in Figure P6.65.

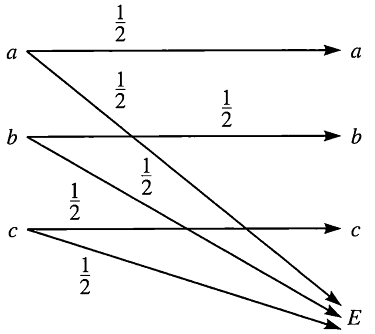


FIGURE P6.65

1. Determine the capacity of this channel.
2. A memoryless exponential source X with probability density function

$$f_X(x) = \begin{cases} 2e^{-2x} & x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

is quantized using a two-level quantizer defined by

$$\hat{X} = q(X) = \begin{cases} 0 & X < 2 \\ 1 & \text{otherwise} \end{cases}$$

Can $\hat{X}$ be reliably transmitted over the channel shown above? Why? (The number of source symbols per second is equal to the number of channel symbols per second.)

6.66 Plot the capacity of an AWGN channel that employs binary antipodal signaling, with optimal bit-by-bit detection at the receiver, as a function of $\mathcal{E}_b/N_0$. On the same axis, plot the capacity of the same channel when binary orthogonal signaling is employed.

6.67 A discrete-time memoryless Gaussian source with mean 0 and variance σ^2 is to be transmitted over a binary symmetric channel with crossover probability p.

1. What is the minimum value of the distortion attainable at the destination (distortion is measured in mean-squared error)?
2. If the channel is a discrete-time memoryless additive Gaussian noise channel with input power P and noise power P_n, what is the minimum attainable distortion?
3. Now assume that the source has the same basic properties but is not memoryless. Do you expect the distortion in transmission over the binary symmetric channel to be decreased or increased? Why?

6.68 Find the capacity of the cascade connection of n binary symmetric channels with the same crossover probability ϵ. What is the capacity when the number of channels goes to infinity?

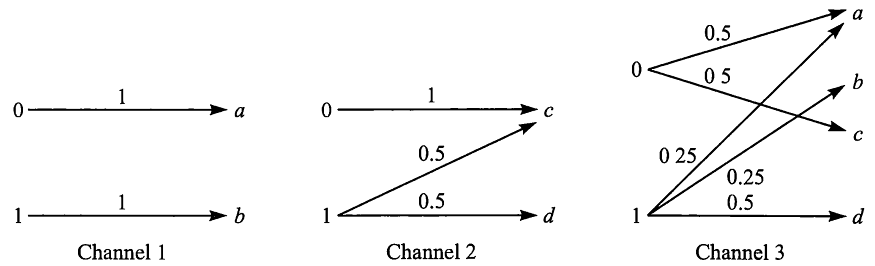
6.69 Channels 1, 2, and 3 are shown in Figure P6.69.

1. Find the capacity of channel 1. What input distribution achieves capacity?
2. Find the capacity of channel 2. What input distribution achieves capacity?
3. Let C denote the capacity of the third channel and C_1 and C_2 represent the capacities of the first and second channels. Which of the following relations holds true and why?

$$C \leq \frac{1}{2}(C_1 + C_2)$$

$$C = \frac{1}{2}(C_1 + C_2)$$

$$C \geq \frac{1}{2}(C_1 + C_2)$$



Channel 1 Channel 2 Channel 3

FIGURE P6.69

6.70 Let C denote the capacity of a discrete memoryless channel with input alphabet $\mathscr{X} = \{x_1, x_2, \ldots, x_N\}$ and output alphabet $\mathscr{Y} = \{y_1, y_2, \ldots, y_M\}$. Show that $C \leq \min\{\log M, \log N\}$.

6.71 The channel C (known as the Z channel) is shown in Figure P6.71.

1. Find the input probability distribution that achieves capacity.
2. What is the input distribution and capacity for the special cases $\epsilon = 0$, $\epsilon = 1$, and $\epsilon = 0.57$?

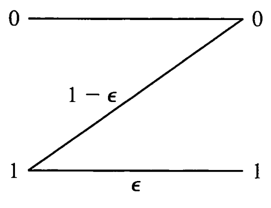


FIGURE P6.71

3. Show that if n such channels are cascaded, the resulting channel will be equivalent to a Z channel with $\epsilon_1 = \epsilon^n$.
4. What is the capacity of the equivalent Z channel when $n \to \infty$?

6.72 Find the capacity of an additive white Gaussian noise channel with a bandwidth 1 MHz, power 10 W, and noise power spectral density $\frac{1}{2}N_0 = 10^{-9}$ W/Hz.

6.73 A Gaussian *memoryless* source is distributed according to $\mathcal{N}(0, 1)$. This source is to be transmitted over a binary symmetric channel with a crossover probability of $\epsilon = 0.1$. For each source output one use of channel is possible. The fidelity measure is squared-error distortion, i.e., $d(x, \hat{x}) = (x - \hat{x})^2$.

1. In the first approach we use the optimum one-dimensional (scalar) quantizer. This results in the following quantization rule

$$Q(x) = \begin{cases} \hat{x} & x > 0 \\ -\hat{x} & x \le 0 \end{cases}$$

   where $\hat{x} = 0.798$ and the resulting distortion is 0.3634. Then $\hat{x}$ and $-\hat{x}$ are represented by 0 and 1 and directly transmitted over the channel (no channel coding). Determine the resulting overall distortion using this approach.
2. In the second approach we use the same quantizer used in part 1, but we allow the use of arbitrarily complex channel coding. How would you determine the resulting distortion in this case, and why?
3. Now assume that after quantization, an arbitrarily complex lossless compression scheme is employed and the output is transmitted over the channel (again using channel coding, as explained in part 2). How would the resulting distortion compare with part 2?
4. If you were allowed to use an arbitrarily complex source and channel coding scheme, what would be the minimum achievable distortion?
5. If the source is Gaussian with the same per-letter statistics (i.e., each letter is $\mathcal{N}(0, 1)$) but the source has *memory* (for instance, a Gauss-Markov source), do you think the distortion you derived in part 4 would increase, decrease, or not change? Why?

6.74 For the channel shown in Figure P6.65:

1. Consider an extension of the channel with inputs $a_1, a_2, \ldots, a_n$, outputs $a_1, a_2, \ldots, a_n, E$, where $P(a_i|a_i) = \frac{1}{2}$, $P(E|a_i) = \frac{1}{2}$, for all $1 \le i \le n$, and all other transition probabilities are zero. What is the capacity of this channel? What is the capacity when $n = 2^m$?
2. If a memoryless binary equiprobable source is transmitted via the channel shown in Figure P6.65, what is the minimum attainable error probability, assuming no limit is imposed on the complexity and delay of the system? (The number of source symbols per second is equal to the number of channel symbols per second.) For what values of n in part 2 can the source be reliably transmitted over the channel?
3. If a Gaussian source distributed according to $\mathcal{N}(m, \sigma^2)$ is transmitted via the channel in part 2, what is the minimum attainable mean-squared distortion in regeneration of this source as a function of n and σ^2? (Again the number of source symbols per second is equal to the number of channel symbols per second, and no limit is imposed on system complexity and delay.)

6.75 Using the expression for the cutoff R_0 for the BSC, given in Equation 6.8–29, plot R_0 as a function of $\mathcal{E}_c/N_0$ for the following binary modulation methods:

1. Antipodal signaling: $p = Q\left(\sqrt{\frac{2\mathcal{E}_c}{N_0}}\right)$
2. Orthogonal signaling: $p = Q\left(\sqrt{\frac{2\mathcal{E}_c}{N_0}}\right)$
3. DPSK: $p = \frac{1}{2}e^{-\mathcal{E}_c/N_0}$

Comment on the difference in performance for the three modulation methods, as given by the cutoff rate.

6.76 Consider the binary-input, ternary-output channel with transition probabilities shown in Figure P6.76, where e denotes an erasure. For the AWGN channel, α and p are defined as

$$\alpha = \frac{1}{\sqrt{\pi N_0}} \int_{-\beta}^{\beta} e^{-\left(x+\sqrt{\mathcal{E}_c}\right)^2/N_0}\, dx$$

$$p = \frac{1}{\sqrt{\pi N_0}} \int_{\beta}^{\infty} e^{-\left(x+\sqrt{\mathcal{E}_c}\right)^2/N_0}\, dx$$

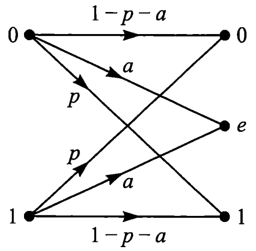


FIGURE P6.76

1. Determine the cutoff rate R_0 as a function of the probabilities α and p.
2. The cutoff rate R_0 depends on the choice of the threshold β through the probabilities α and p. For any $\mathcal{E}_c/N_0$, the value of β that maximizes R_0 can be determined by trial and error. For example, it can be shown that for $\mathcal{E}_c/N_0$ below 0 dB, $\beta_{\text{opt}} = 0.65\sqrt{\frac{1}{2}N_0}$; for $1 \le \mathcal{E}_c/N_0 \le 10$, β_{opt} varies approximately linearly between $0.65\sqrt{\frac{1}{2}N_0}$ and $\sqrt{\frac{1}{2}N_0}$. By using $\beta = 0.65\sqrt{\frac{1}{2}N_0}$ for the entire range of $\mathcal{E}_c/N_0$, plot R_0 versus $\mathcal{E}_c/N_0$ and compare this result with R_0 for an unquantized (continuous) output channel.

6.77 Show that for M-ary PSK signaling the cutoff rate R_0 is given by

$$\begin{aligned} R_0 &= \log_2 M - \log_2 \left[\sum_{k=0}^{M-1} e^{-\|s_0 - s_k\|^2/4N_0}\right] \\ &= \log_2 M - \log_2 \left[\sum_{k=0}^{M-1} e^{-(\mathcal{E}_c/N_0)\sin^2(\pi k/M)}\right] \end{aligned}$$

Plot R_0 as a function of $\mathcal{E}_c/N_0$ for $M = 2, 4, 8$, and 16.

6.78 A discrete-time additive *non-Gaussian* noise channel is described by the input-output relation

$$y_i = x_i + n_i$$

where n_i represents a sequence of iid noise random variables with probability density function

$$p(n) = \frac{1}{2} e^{-|n|}$$

and x_i can take ±1 with equal probability, where i represents the time index.

1. Determine the cutoff rate R_0 for this channel.
2. Assume that this channel is used with optimal hard decision decoding at the output. What is the crossover probability of the resulting BSC channel?
3. What is the cutoff rate in part 2?

6.79 Show that the cutoff rate for an M-ary orthogonal signaling system where each signal has energy E and the channel is AWGN with noise power spectral density of $\frac{1}{2}N_0$ can be expressed as

$$R_0 = \log_2 M - \log_2 \left[1 + (M-1) \left(\int_{-\infty}^{\infty} p_n(y - \sqrt{\mathcal{E}}) p_n(y)\, dy \right)^2 \right]$$

where $p_n(\cdot)$ represents the PDF of an $\mathcal{N}(0, \frac{1}{2}N_0)$ random variable. Conclude that the above expression is simplified as

$$R_0 = \log_2 \left[\frac{M}{1 + (M-1)\, e^{-\mathcal{E}/N_0}} \right]$$

7

Linear Block Codes

We have studied the performance of different signaling methods when transmitted through an AWGN channel in Chapter 4. In particular we have seen how the error probability of each signaling method is related to the SNR per bit. In that chapter we were mainly concerned with the case where M possible messages are sent by transmitting one of the M possible waveforms, rather than blocks of channel inputs. We also introduced criteria for comparing power and bandwidth efficiency of different signaling schemes. The power efficiency is usually measured in terms of the required SNR per bit to achieve a certain error probability. The lower the required SNR per bit, the more power-efficient the system is. The bandwidth efficiency of the system is measured by the spectral bit rate $r = R/W$ which determines how many bits per second can be transmitted in 1 Hz of bandwidth. Systems with high spectral bit rate are highly bandwidth-efficient systems. We also saw that there is a trade-off between bandwidth and power efficiency. Modulation schemes such as QAM are highly bandwidth-efficient, and signaling schemes such as orthogonal signaling are power-efficient at the expense of high bandwidth demand.

In Chapter 6 we saw that reliable communication over a noisy channel is possible if the transmission rate is less than channel capacity. Reliable communication is made possible through *channel coding*, i.e., assigning messages to *blocks of channel inputs* and *using only a subset of all possible blocks*. In Chapter 6 we did not study specific mappings between messages and channel input sequences. Both channel capacity C and channel cutoff rate R_0 were presented using *random coding*. In random coding we do not find the best mapping from the message set to channel input sequences and analyze the performance of that mapping; rather we average the error probability over all possible mappings and show that if the transmission rate is less than channel capacity, the *ensemble average* of the error probability, averaged over all possible mappings, goes to zero as the block length increases. From this we concluded that there must exist at least one mapping among all mappings for which the error probability goes to zero as the block length increases. The original proof of the channel coding theorem, presented by Shannon in 1948, was based on random coding, and hence was not constructive in the sense that it proved only the existence of good codes but did not provide any method for their design. Of course, based on the idea of random coding, one can argue that there is a good chance that a randomly generated code is a good code. The problem, however, is that the decoding of a randomly generated code when the codeword sequences are long becomes extremely complex, thus making its use in practical systems impossible. The development of coding theory in the decades after 1948 has been focused on designing coding schemes that have sufficient structure to make their decoding practical and at the same time close the gap between an uncoded system and the bounds derived by Shannon. In Chapter 6 we also derived a fundamental relation between r, the spectral bit rate, and $\frac{E_b}{N_0}$, the SNR per bit of an ideal communication system given by

$$\frac{E_b}{N_0} > \frac{2^r - 1}{r}$$

By comparing the bandwidth and power efficiency of a given system with the bound given in this equation, we can see how much that system can be improved.

Our focus in this chapter and Chapter 8 is on channel coding schemes with manageable decoding algorithms that are used to improve performance of communication systems over noisy channels. This chapter is devoted to block codes whose construction is based on familiar algebraic structures such as groups, rings, and fields. In Chapter 8 we will study coding schemes that are best represented in terms of graphs and trellises.

7.1
BASIC DEFINITIONS

Channel codes can be classified into two major classes, *block codes* and *convolutional codes*. In block codes one of the $M = 2^k$ messages, each representing a binary sequence of length k, called the *information sequence*, is mapped to a binary sequence of length n, called the *codeword*, where $n > k$. The codeword is usually transmitted over the communication channel by sending a sequence of n binary symbols, for instance, by using BPSK. QPSK and BFSK are other types of signaling schemes frequently used for transmission of a codeword. Block coding schemes are memoryless. After a codeword is encoded and transmitted, the system receives a new set of k information bits and encodes them using the mapping defined by the coding scheme. The resulting codeword depends only on the current k information bits and is independent of all the codewords transmitted before.

Convolutional codes are described in terms of finite-state machines. In these codes, at each time instance i, k information bits enter the encoder, causing n binary symbols generated at the encoder output and changing the state of the encoder from σ_{i-1} to σ_i. The set of possible states is finite and denoted by Σ. The n binary symbols generated at the encoder output and the next state σ_i depend on the k input bits as well as σ_{i-1}. We can represent a convolutional code by a shift register of length Kk as shown in Figure 7.1–1.

At each time instance, k bits enter the encoder and the contents of the shift register are shifted to the right by k memory elements. The contents of the rightmost k elements of the shift register leave the encoder. After the k bits have entered the shift register,

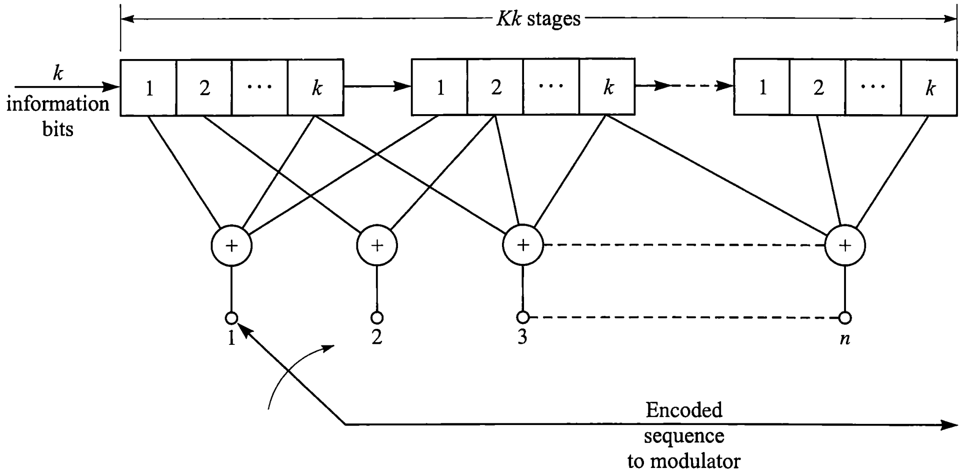


FIGURE 7.1–1
A convolutional encoder.

the n adders add the contents of the memory elements they are connected to (modulo-2 addition) thus generating the code sequence of length n which is sent to the modulator. The state of this convolutional code is given by the contents of the first $(K - 1)k$ elements of the shift register.

The *code rate* of a block or convolutional code is denoted by R_c and is given by

$$R_c = \frac{k}{n} \tag{7.1–1}$$

The rate of a code represents the number of information bits sent in transmission of a binary symbol over the channel. The unit of R_c is information bits per transmission. Since generally $n > k$, we have $R_c < 1$.

Let us assume that a codeword of length n is transmitted using an N-dimensional constellation of size M, where M is assumed to be a power of 2 and $L = \frac{n}{\log_2 M}$ is assumed to be an integer representing the number of M-ary symbol transmitted per codeword. If the symbol duration is T_s, then the transmission time for k bits is $T = LT_s$ and the transmission rate is given by

$$R = \frac{k}{LT_s} = \frac{k}{n} \times \frac{\log_2 M}{T_s} = R_c \frac{\log_2 M}{T_s} \text{ bits/s} \tag{7.1–2}$$

The dimension of the space of the encoded and modulated signals is LN, and using the dimensionality theorem as stated in Equation 4.6–5 we conclude that the minimum required transmission bandwidth is given by

$$W = \frac{N}{2T_s} = \frac{RN}{2R_c \log_2 M} \text{ bits/s} \tag{7.1–3}$$

and from Equation 7.1–3, the resulting spectral bit rate is given by

$$r = \frac{R}{W} = \frac{2 \log_2 M}{N} R_c \tag{7.1–4}$$

These equations indicate that compared with an uncoded system that uses the same modulation scheme, the bit rate is changed by a factor of R_c and the bandwidth is changed by a factor of $1/R_c$, i.e., there is a decrease in rate and an increase in bandwidth.

If the average energy of the constellation is denoted by $\mathcal{E}_{\text{av}}$, then the energy per codeword $\mathcal{E}$, is given by

$$\mathcal{E} = L\mathcal{E}_{\text{av}} = \frac{n}{\log_2 M}\mathcal{E}_{\text{av}} \tag{7.1–5}$$

and $\mathcal{E}_c$, energy per component of the codeword, is given by

$$\mathcal{E}_c = \frac{\mathcal{E}}{n} = \frac{\mathcal{E}_{\text{av}}}{\log_2 M} \tag{7.1–6}$$

The energy per transmitted bit is denoted by $\mathcal{E}_b$ and can be found from

$$\mathcal{E}_b = \frac{\mathcal{E}}{k} = \frac{\mathcal{E}_{\text{av}}}{R_c \log_2 M} \tag{7.1–7}$$

From Equations 7.1–6 and 7.1–7 we conclude that

$$\mathcal{E}_c = R_c\mathcal{E}_b \tag{7.1–8}$$

The transmitted power is given by

$$P = \frac{\mathcal{E}}{LT_s} = \frac{\mathcal{E}_{\text{av}}}{T_s} = R\frac{\mathcal{E}_{\text{av}}}{R_c \log_2 M} = R\mathcal{E}_b \tag{7.1–9}$$

Modulation schemes frequently used with coding are BPSK, BFSK, and QPSK. The minimum required bandwidth and the resulting spectral bit rates for these modulation schemes[†] are given below:

$$\text{BPSK}: \begin{cases} W = \frac{R}{R_c} \\ r = R_c \end{cases} \qquad \text{BFSK}: \begin{cases} W = \frac{R}{R_c} \\ r = R_c \end{cases} \qquad \text{QPSK}: \begin{cases} W = \frac{R}{2R_c} \\ r = 2R_c \end{cases} \tag{7.1–10}$$

7.1–1 The Structure of Finite Fields

To further explore properties of block codes, we need to introduce the notion of a *finite field* and its main properties. Simply stated, a field is a collection of objects that can be added, subtracted, multiplied; and divided. To define fields, we begin by defining *Abelian groups*. An Abelian group is a set with a binary operation that has the basic properties of addition. A set G and a binary operation denoted by $+$ constitute an Abelian group if the following properties hold:

1. The operation $+$ is commutative; i.e., for any $a, b \in G$, $a + b = b + a$.
2. The operation $+$ is associative; i.e., for any $a, b, c \in G$, we have $(a + b) + c = a + (b + c)$.

[†] BPSK is assumed to be transmitted as a double-sideband signal.

TABLE 7.1–1
Addition and Multiplication Tables for GF(2)

$+$	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

3. The operation $+$ has an *identity element* denoted by 0 such that for any $a \in G$, $a + 0 = 0 + a = a$.
4. For any $a \in G$ there exists an element $-a \in G$ such that $a + (-a) = (-a) + a = 0$. The element $-a$ is called the (additive) *inverse* of a.

An Abelian group is usually denoted by $\{G, +, 0\}$.

A finite field or *Galois field*† is a finite set F with two binary operations, addition and multiplication, denoted, respectively, by $+$ and $\cdot$, satisfying the following properties:

1. $\{F, +, 0\}$ is an Abelian group.
2. $\{F - \{0\}, \cdot, 1\}$ is an Abelian group; i.e., the nonzero elements of the field constitute an Abelian group under multiplication with an identity element denoted by "1". The multiplicative inverse of $a \in F$ is denoted by a^{-1}.
3. Multiplication is distributive with respect to addition: $a \cdot (b + c) = (b + c) \cdot a = a \cdot b + a \cdot c$.

A field is usually denoted by $\{F, +, \cdot\}$. It is clear that $\mathbb{R}$, the set of real numbers, is a field (but not a finite field) with ordinary addition and multiplication. The set $F = \{0, 1\}$ with modulo-2 addition and multiplication is an example of a Galois (finite) field. This field is called the binary field and is denoted by GF(2). The addition and multiplication tables for this field are given in Table 7.1–1.

Characteristic of a Field and the Ground Field

A fundamental theorem of algebra states that a Galois field with q elements, denoted by GF(q), exists if and only if $q = p^m$, where p is a prime and m is a positive integer. It can also be proved that when GF(q) exists, it is unique up to isomorphism. This means that any two Galois fields of the same size can be obtained from each other after renaming the elements. For the case of $q = p$, the Galois field can be denoted by GF(p) $= \{0, 1, 2, \ldots, p - 1\}$ with modulo-p addition and multiplication. For instance GF(5) $= \{0, 1, 2, 3, 4\}$ is a finite field with modulo-5 addition and multiplication. When $q = p^m$, the resulting Galois field is called an *extension field* of GF(p). In this case GF(p) is called the *ground field* of GF(p^m), and p is called the *characteristic* of GF(p^m).

---

†Named after French mathematician Évariste Galois (1811–1832).

Polynomials over Finite Fields

To study the structure of extension fields, we need to define polynomials over GF(p). A polynomial of degree m over GF(p) is a polynomial

$$g(X) = g_0 + g_1 X + g_2 X^2 + \cdots + g_m X^m \tag{7.1–11}$$

where g_i, $0 \leq i \leq m$, are elements of GF(p) and $g_m \neq 0$. Addition and multiplication of polynomials follow standard addition and multiplication rules of ordinary polynomials except that addition and multiplication of the coefficients are done modulo-p. If $g_m = 1$, the polynomial is called *monic*. If a polynomial of degree m over GF(p) cannot be written as the product of two polynomials of lower degrees over the same Galois field, then the polynomial is called an *irreducible* polynomial. For instance, $X^2 + X + 1$ is an irreducible polynomial over GF(2), whereas $X^2 + 1$ is not irreducible over GF(2) because $X^2 + 1 = (X + 1)^2$. A polynomial that is both monic and irreducible is called a *prime* polynomial. A fundamental result of algebra states that a polynomial of degree m over GF(p) has m roots (some may be repeated), but the roots are not necessarily in GF(p). In general, the roots are in some extension field of GF(p).

The Structure of Extension Fields

From the above definitions it is clear that there exist p^m polynomials of degree less than m; in particular these polynomials include two special polynomials $g(X) = 0$ and $g(X) = 1$. Now let us assume that $g(X)$ is a prime (monic and irreducible) polynomial of degree m and consider the set of all polynomials of degree less than m over GF(p) with ordinary addition and with polynomial multiplication modulo-$g(X)$. It can be shown that the set of these polynomials with the addition and multiplication operations defined above is a Galois field with p^m elements.

EXAMPLE 7.1–1. We know that $X^2 + X + 1$ is prime over GF(2); therefore this polynomial can be used to construct GF(2^2) = GF(4). Let us consider all polynomials of degree less than 2 over GF(2). These polynomials are 0, 1, X, and $X + 1$ with addition and multiplication tables given in Table 7.1–2. Note that the multiplication rule basically entails multiplying the two polynomials, dividing the product by $g(X) = X^2 + X + 1$, and finding the remainder. This is what is meant by multiplying modulo-$g(X)$. It is interesting to note that all nonzero elements of GF(4) can be written as powers of X; i.e, $X = X^1$, $X + 1 = X^2$, and $1 = X^3$.

TABLE 7.1–2
Addition and Multiplication Table for GF(4)

$+$	0	1	X	$X+1$
0	0	1	X	$X+1$
1	1	0	$X+1$	X
X	X	$X+1$	0	1
$X+1$	$X+1$	X	1	0

$\cdot$	0	1	X	$X+1$
0	0	0	0	0
1	0	1	X	$X+1$
X	0	X	$X+1$	1
$X+1$	0	$X+1$	1	X

TABLE 7.1–3
Multiplication Table for GF(8)

$\cdot$	0	1	X	$X+1$	X^2	X^2+1	X^2+X	X^2+X+1
0	0	0	0	0	0	0	0	0
1	0	1	X	$X+1$	X^2	X^2+1	X^2+X	X^2+X+1
X	0	X	X^2	X^2+X	$X+1$	1	X^2+X+1	X^2+1
$X+1$	0	$X+1$	X^2+X	X^2+1	X^2+X+1	X^2	1	X
X^2	0	X^2	$X+1$	X^2+X+1	X^2+X	X	X^2+1	1
X^2+1	0	X^2+1	1	X^2	X	$X+2+X+1$	$X+1$	X^2+X
X^2+X	0	X^2+X	X^2+X+1	1	X^2+1	$X+1$	X	X^2
X^2+X+1	0	X^2+X+1	X^2+1	X	1	X^2+X	X^2	$X+1$

EXAMPLE 7.1–2. To generate GF(2^3), we can use either of the two prime polynomials $g_1(X) = X^3 + X + 1$ or $g_2(X) = X^3 + X^2 + 1$. If $g(X) = X^3 + X + 1$ is used, the multiplication table for GF(2^3) is given by Table 7.1–3. The addition table has a trivial structure. Here again note that $X^1 = X$, $X^2 = X^2$, $X^3 = X + 1$, $X^4 = X^2 + X$, $X^5 = X^2 + X + 1$, $X^6 = X^2 + 1$, and $X^7 = 1$. In other words, all nonzero elements of GF(8) can be written as powers of X. The nonzero elements of the field can be expressed either as polynomials of degree less than 3 or, equivalently, as X^i for $1 \leq i \leq 7$. A third method for representing the field elements is to write coefficients of the polynomial as a vector of length 3. The representation of the form X^i is the appropriate representation when multiplying field elements since $X^i \cdot X^j = X^{i+j}$, where $i + j$ should be reduced modulo-7 because $X^7 = 1$. The polynomial and vector representations of field elements are more appropriate when adding field elements. A table of the three representations of field elements is given in Table 7.1–4. For instance, to multiply $X^2 + X + 1$ and $X^2 + 1$, we use their power representation as X^5 and X^6 and we have $(X^2 + X + 1)(X^2 + 1) = X^{11} = X^4 = X^2 + X$.

TABLE 7.1–4
Three Representations for GF(8) Elements

Power	Polynomial	Vector
—	0	000
$X^0 = X^7$	1	001
X^1	X	010
X^2	X^2	100
X^3	$X+1$	011
X^4	X^2+X	110
X^5	X^2+X+1	111
X^6	X^2+1	101

Primitive Elements and Primitive Polynomials

For any nonzero element $\beta \in \mathrm{GF}(q)$, the smallest value of i such that $\beta^i = 1$ is called the *order* of β. It is shown in Problem 7.1 that for any nonzero $\beta \in \mathrm{GF}(q)$ we have $\beta^{q-1} = 1$; therefore the order of β is at most equal to $q - 1$. A nonzero element of GF(q) is called a *primitive element* if its order is $q - 1$. We observe that in both Examples 7.1–1 and 7.1–2, X is a primitive element. Primitive elements have the property that their powers generate all nonzero elements of the Galois field. Primitive elements are not unique; for instance, the reader can verify that in the GF(8) of Example 7.1–2, X^2 and $X + 1$ are both primitive elements; however, $1 \in \mathrm{GF}(8)$ is not primitive since $1^1 = 1$.

Since there are many prime polynomials of degree m, there are many constructs of GF(p^m) which are all isomorphic; i.e., each can be obtained from another by renaming the elements. It is desirable that X be a primitive element of the Galois field GF(p^m), since in this case all nonzero elements of the field can be expressed simply as powers of X as was shown in Table 7.1–4 for GF(8). If GF(p^m), generated by $g(X)$, is such that in this field X is a primitive element, then the polynomial $g(X)$ is called a *primitive polynomial*. It can be shown that primitive polynomials exist for any degree m; and therefore, for any positive integer m and any prime p, it is possible to generate GF(p^m) such that in this field X is primitive, i.e., all nonzero elements can be written as X^i, $0 \le i < p^m - 1$. We always assume that Galois fields are constructed using primitive polynomials.

EXAMPLE 7.1–3. Polynomials $g_1(X) = X^4 + X + 1$ and $g_2(X) = X^4 + X^3 + X^2 + X + 1$ are two prime polynomials of degree 4 over GF(2) that can be used to generate GF(2^4). However, in the Galois field generated by $g_1(X)$, X is a primitive element, hence $g_1(X)$ is a primitive polynomial, but in the field generated by $g_2(X)$, X is not primitive; in fact in this field $X^5 = 1$ since $X^5 + 1 = (X + 1)g_2(X)$. Therefore, $g_2(X)$ is not a primitive polynomial.

It can be shown that any prime polynomial $g(X)$ of degree m over GF(p) divides $X^{p^m-1} + 1$. However, it is possible that $g(X)$ divides $X^i + 1$ for some $i < p^m - 1$ as well. For instance, $X^4 + X^3 + X^2 + X + 1$ divides $X^{15} + 1$, but it also divides $X^5 + 1$. It can be shown that if a prime polynomial $g(X)$ has the property that the smallest integer i for which $g(X)$ divides $X^i + 1$ is $i = p^m - 1$, then $g(X)$ is primitive. This means that we have two equivalent definitions for a primitive polynomial. The first definition states that a primitive polynomial $g(X)$ is a prime polynomial of degree m such that if GF(p^m) is constructed based on $g(X)$, in the resulting field X is a primitive element. The second definition states that $g(X)$, a prime polynomial of degree m, is primitive if $g(X)$ does not divide $X^i + 1$ for any $i < p^m - 1$. All roots of a primitive polynomial of degree m are primitive elements of GF(p^m). Primitive polynomials are usually tabulated for different values of m. Table 7.1–5 gives some primitive polynomials for $2 \le m \le 12$.

EXAMPLE 7.1–4. GF(16) can be constructed using $g(X) = X^4 + X + 1$. If α is a root of $g(X)$, then α is a primitive element of GF(16) and all nonzero elements of GF(16) can be written as α^i for $0 \le i < 15$ with $\alpha^{15} = \alpha^0 = 1$. Table 7.1–6 presents elements of GF(16) as powers of α, as polynomials in α, and finally as binary vectors of length 4. Note that $\beta = \alpha^3$ is a nonprimitive element in this field since $\beta^5 = \alpha^{15} = 1$; i.e., the order of β is 5. It is clearly seen that α^6, α^{12}, and α^9 are also elements of order 5, whereas α^5 and α^{10} are elements of order 3. Primitive elements of this field are α, α^2, α^4, α^8, α^7, α^{14}, α^{13}, and α^{11}.

TABLE 7.1–5
Primitive Polynomials of Orders 2 through 12

m	$g(X)$
2	X^2+X+1
3	X^3+X+1
4	X^4+X+1
5	X^5+X^2+1
6	X^6+X+1
7	X^7+X^3+1
8	$X^8+X^4+X^3+X^2+1$
9	X^9+X^4+1
10	$X^{10}+X^3+1$
11	$X^{11}+X^2+1$
12	$X^{12}+X^6+X^4+X+1$

Minimal Polynomials and Conjugate Elements

The minimal polynomial of a field element is the lowest-degree monic polynomial over the ground field that has the element as its root. Let β be a nonzero element of GF(2^m). Then the *minimal polynomial* of β, denoted by $\phi_\beta(X)$, is a monic polynomial of lowest degree with coefficients in GF(2) such that β is a root of $\phi_\beta(X)$, i.e., $\phi_\beta(\beta) = 0$. Obviously $\phi_\beta(X)$ is a prime polynomial over GF(2) and divides any other polynomial over GF(2) that has a root at β; i.e., if $f(X)$ is any polynomial over GF(2) such that

TABLE 7.1–6
Elements of GF(16)

Power	Polynomial	Vector
—	0	0000
$\alpha^0=\alpha^{15}$	1	0001
α^1	α	0010
α^2	α^2	0100
α^3	α^3	1000
α^4	$\alpha+1$	0011
α^5	$\alpha^2+\alpha$	0110
α^6	$\alpha^3+\alpha^2$	1100
α^7	$\alpha^3+\alpha+1$	1011
α^8	α^2+1	0101
α^9	$\alpha^3+\alpha$	1010
α^{10}	$\alpha^2+\alpha+1$	0111
α^{11}	$\alpha^3+\alpha^2+\alpha$	1110
α^{12}	$\alpha^3+\alpha^2+\alpha+1$	1111
α^{13}	$\alpha^3+\alpha^2+1$	1101
α^{14}	α^3+1	1001

$f(\beta) = 0$, then $f(X)$ can be factorized as $f(X) = a(X)\phi_\beta(X)$. In the following paragraph we see how to obtain the minimal polynomial of a field element.

Since $\beta \in \mathrm{GF}(2^m)$ and $\beta \neq 0$, we know that $\beta^{2^m-1} = 1$. However, it is possible that for some integer $\ell < m$ we have $\beta^{2^\ell-1} = 1$. For instance, in GF(16) if $\beta = \alpha^5$, then $\beta^3 = \beta^{2^2-1} = 1$; therefore for this β we have $\ell = 2$. It can be shown that for any $\beta \in \mathrm{GF}(2^m)$, the minimal polynomial $\phi_\beta(X)$ is given by

$$\phi_\beta(X) = \prod_{i=0}^{\ell-1} \left(X + \beta^{2^i}\right) \tag{7.1–12}$$

where ℓ is the smallest integer such that $\beta^{2^\ell-1} = 1$. The roots of $\phi_\beta(X)$, i.e., elements of the form β^{2^i}, $1 < i \leq \ell - 1$, are called *conjugates* of β. It can be shown that all conjugates of an element of a finite field have the same order. This means that conjugates of primitive elements are also primitive. We add here that although all conjugates have the same order, this does not mean that all elements of the same order are necessarily conjugates. All elements of the finite field that are conjugates of each other are said to belong to the same *conjugacy class*. Therefore to find the minimal polynomial of $\beta \in \mathrm{GF}(q)$, we take the following steps:

1. Find the conjugacy class of β, i.e., all elements of the form β^{2^i} for $0 \leq i \leq \ell - 1$ where ℓ is the smallest positive integer such that $\beta^{2^\ell} = \beta$.
2. Find $\phi_\beta(X)$ as a monic polynomial whose roots are in the conjugacy class of β. This is done by using Equation 7.1–12.

The $\phi_\beta(X)$ obtained by this procedure is guaranteed to be a prime polynomial with coefficients in GF(2).

EXAMPLE 7.1–5. To find the minimal polynomial of $\beta = \alpha^5$ in GF(16), we observe that $\beta^4 = \alpha^{20} = \alpha^5 = \beta$. Hence, $\ell = 2$, and the conjugacy class is $\{\beta, \beta^2\}$. Therefore,

$$\begin{aligned}\phi_\beta(X) &= \prod_{i=0}^{1} \left(X + \beta^{2^i}\right)\\ &= (X + \beta)(X + \beta^2)\\ &= (X + \alpha^5)(X + \alpha^{10})\\ &= X^2 + (\alpha^5 + \alpha^{15})X + \alpha^{15}\\ &= X^2 + X + 1\end{aligned} \tag{7.1–13}$$

For $\gamma = \alpha^3$ we have $\ell = 4$ and the conjugacy class is $\{\gamma, \gamma^2, \gamma^4, \gamma^8\}$. Therefore,

$$\begin{aligned}\phi_\gamma(X) &= \prod_{i=0}^{3} \left(X + \gamma^{2^i}\right)\\ &= (X + \gamma)(X + \gamma^2)(X + \gamma^4)(X + \gamma^8)\\ &= (X + \alpha^3)(X + \alpha^6)(X + \alpha^{12})(X + \alpha^9)\\ &= X^4 + X^3 + X^2 + X + 1\end{aligned} \tag{7.1–14}$$

To find the minimal polynomial of α, we note that $\alpha^{16} = \alpha$, hence $\ell = 4$ and the conjugacy class is $\{\alpha, \alpha^2, \alpha^4, \alpha^8\}$. The resulting minimal polynomial is

$$\begin{aligned} \phi_\alpha(X) &= \prod_{i=0}^{3} \left(X + \alpha^{2^i}\right) \\ &= (X+\alpha)(X+\alpha^2)(X+\alpha^4)(X+\alpha^8) \\ &= X^4 + X + 1 \end{aligned} \qquad (7.1\text{–}15)$$

For $\delta = \alpha^7$ we again have $\ell = 4$, and the conjugacy class is $\{\delta, \delta^2, \delta^4, \delta^8\}$. The minimal polynomial is

$$\begin{aligned} \phi_\delta(X) &= \prod_{i=0}^{3} \left(X + \delta^{2^i}\right) \\ &= (X+\alpha^7)(X+\alpha^{14})(X+\alpha^{13})(X+\alpha^{11}) \\ &= X^4 + X^3 + 1 \end{aligned} \qquad (7.1\text{–}16)$$

Note that α and δ are both primitive elements, but they belong to two different conjugacy classes and thus have different minimal polynomials.

We conclude our discussion of Galois field properties by observing that all the p^m elements of GF(p^m) are the roots of the equation

$$X^{p^m} - X = 0 \qquad (7.1\text{–}17)$$

or equivalently, all nonzero elements of GF(p^m) are the roots of

$$X^{p^m-1} - 1 = 0 \qquad (7.1\text{–}18)$$

This means that the polynomial $X^{2^m-1} - 1$ can be uniquely factored over GF(2) into the product of the minimal polynomials corresponding to the conjugacy classes of nonzero elements of GF(2^m). In fact $X^{2^m-1} - 1$ can be factorized over GF(2) as the product of all prime polynomials over GF(2) whose degree divides m. For more details on the structure of finite fields and the proofs of the properties we covered here, the reader is referred to MacWilliams and Sloane (1977), Wicker (1995), and Blahut (2003).

7.1–2 Vector Spaces

A vector a space over a *field of scalars* $\{F, +, \cdot\}$ is an Abelian group $\{V, +, \mathbf{0}\}$ whose elements are denoted by boldface symbols such as $\boldsymbol{v}$ and called *vectors*, with vector addition $+$ and identity element $\mathbf{0}$; and an operation called *scalar multiplication* for each $c \in F$ and each $\boldsymbol{v} \in V$ that is denoted by $c \cdot \boldsymbol{v}$ such that the following properties are satisfied:

1. $c \cdot \boldsymbol{v} \in V$
2. $c \cdot (\boldsymbol{v}_1 + \boldsymbol{v}_2) = c \cdot \boldsymbol{v}_1 + c \cdot \boldsymbol{v}_2$
3. $c_1 \cdot (c_2 \cdot \boldsymbol{v}) = (c_1 \cdot c_2) \cdot \boldsymbol{v}$
4. $(c_1 + c_2) \cdot \boldsymbol{v} = c_1 \cdot \boldsymbol{v} + c_2 \cdot \boldsymbol{v}$
5. $1 \cdot \boldsymbol{v} = \boldsymbol{v}$

It can be easily shown that the following properties are satisfied:

1. $0 \cdot \boldsymbol{v} = \mathbf{0}$
2. $c \cdot \mathbf{0} = \mathbf{0}$
3. $(-c) \cdot \boldsymbol{v} = c \cdot (-\boldsymbol{v}) = -(c \cdot \boldsymbol{v})$

We will be mainly dealing with vector spaces over the scalar field GF(2). In this case a vector space V is a collection of binary n-tuples such that if $\boldsymbol{v}_1, \boldsymbol{v}_2 \in V$, then $\boldsymbol{v}_1 + \boldsymbol{v}_2 \in V$, where $+$ denotes componentwise binary addition, or componentwise EXCLUSIVE-OR operation. Note that since we can choose $\boldsymbol{v}_2 = \boldsymbol{v}_1$, we have $\mathbf{0} \in V$.

7.2
GENERAL PROPERTIES OF LINEAR BLOCK CODES

A q-ary block code $\mathcal{C}$ consists of a set of M vectors of length n denoted by $\boldsymbol{c}_m = (c_{m1}, c_{m2}, \ldots, c_{mn})$, $1 \leq m \leq M$, and called *codewords* whose components are selected from an alphabet of q symbols, or elements. When the alphabet consists of two symbols, 0 and 1, the code is a *binary code*. It is interesting to note that when q is a power of 2, i.e., $q = 2^b$ where b is a positive integer, each q-ary symbol has an equivalent binary representation consisting of b bits; thus, a nonbinary code of block length N can be mapped into a binary code of block length $n = bN$.

There are 2^n possible codewords in a binary block code of length n. From these 2^n codewords, we may select $M = 2^k$ codewords ($k < n$) to form a code. Thus, a block of k information bits is mapped into a codeword of length n selected from the set of $M = 2^k$ codewords. We refer to the resulting block code as an (n, k) code, with rate $R_c = k/n$. More generally, in a code having q symbols, there are q^n possible codewords. A subset of $M = q^k$ codewords may be selected to transmit k-symbol blocks of information.

Besides the code rate parameter R_c, an important parameter of a codeword is its *weight*, which is simply the number of nonzero elements that it contains. In general, each codeword has its own weight. The set of all weights in a code constitutes the *weight distribution* of the code. When all the M codewords have equal weight, the code is called a *fixed-weight code* or a *constant-weight code*.

A subset of block codes, called linear block codes, is particularly well studied during the last few decades. The reason for the popularity of linear block codes is that linearity guarantees easier implementation and analysis of these codes. In addition, it is remarkable that the performance of the class of linear block codes is similar to the performance of the general class of block codes. Therefore, we can limit our study to the subclass of linear block codes without sacrificing system performance.

A *linear block code* $\mathcal{C}$ is a k-dimensional subspace of an n-dimensional space which is usually called an (n, k) code. For binary codes, it follows from Problem 7.11 that a linear block code is a collection of 2^k binary sequences of length n such that for any two codewords $\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C}$ we have $\boldsymbol{c}_1 + \boldsymbol{c}_2 \in \mathcal{C}$. Obviously, $\mathbf{0}$ is a codeword of any linear block code.

7.2–1 Generator and Parity Check Matrices

In a linear block code, the mapping from the set of $M = 2^k$ information sequences of length k to the corresponding 2^k codewords of length n can be represented by a $k \times n$ matrix $\boldsymbol{G}$ called the *generator matrix* as

$$\boldsymbol{c}_m = \boldsymbol{u}_m \boldsymbol{G}, \qquad 1 \leq m \leq 2^k \tag{7.2–1}$$

where $\boldsymbol{u}_m$ is a binary vector of length k denoting the information sequence and $\boldsymbol{c}_m$ is the corresponding codeword. The rows of $\boldsymbol{G}$ are denoted by $\boldsymbol{g}_i, 1 \leq i \leq k$, denoting the codewords corresponding to the information sequences $(1, 0, \ldots, 0)$, $(0, 1, 0, \ldots, 0), \ldots, (0, \ldots, 0, 1)$.

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{g}_1 \\ \boldsymbol{g}_2 \\ \vdots \\ \boldsymbol{g}_k \end{bmatrix} \tag{7.2–2}$$

and hence,

$$\boldsymbol{c}_m = \sum_{i=1}^{k} u_{mi} \boldsymbol{g}_i \tag{7.2–3}$$

where the summation is in GF(2), i.e., modulo-2 summation.

From Equation 7.2–2 it is clear that the set of codewords of $\mathcal{C}$ is exactly the set of linear combinations of the rows of $\boldsymbol{G}$, i.e., the row space of $\boldsymbol{G}$. Two linear block codes $\mathcal{C}_1$ and $\mathcal{C}_2$ are called *equivalent* if the corresponding generator matrices have the same row space, possibly after a permutation of columns.

If the generator matrix $\boldsymbol{G}$ has the following structure

$$\boldsymbol{G} = [\boldsymbol{I}_k \mid \boldsymbol{P}] \tag{7.2–4}$$

where $\boldsymbol{I}_k$ is a $k \times k$ identity matrix and $\boldsymbol{P}$ is a $k \times (n-k)$ matrix, the resulting linear block code is called *systematic*. In systematic codes the first k components of the codeword are equal to the information sequence, and the following $n - k$ components, called the *parity check bits*, provide the redundancy for protection against errors. It can be shown that any linear block code has a systematic equivalent; i.e., its generator matrix can be put in the form given by Equation 7.2–4 by elementary row operations and column permutation.

Since $\mathcal{C}$ is a k-dimensional subspace of the n-dimensional binary space, its orthogonal complement, i.e., the set of all n-dimensional binary vectors that are orthogonal to the the codewords of $\mathcal{C}$, is an $(n - k)$-dimensional subspace of the n-dimensional space, and therefore it defines an $(n, n - k)$ code which is denoted by $\mathcal{C}^{\perp}$ and is called the *dual code* of $\mathcal{C}$. The generator matrix of the dual code is an $(n - k) \times n$ matrix whose rows are orthogonal to the rows of $\boldsymbol{G}$, the generator matrix of $\mathcal{C}$. The generator matrix of the dual code is called the *parity check matrix* of the original code $\mathcal{C}$ and is denoted by $\boldsymbol{H}$. Since any codeword of $\mathcal{C}$ is orthogonal to all rows of $\boldsymbol{H}$, we conclude that for all $\boldsymbol{c} \in \mathcal{C}$

$$\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0} \tag{7.2–5}$$

Also if for some binary n-dimensional vector $\boldsymbol{c}$ we have $\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0}$, then $\boldsymbol{c}$ belongs to the orthogonal complement of $\boldsymbol{H}$, i.e., $\boldsymbol{c} \in \mathcal{C}$. Therefore, a necessary and sufficient condition for $\boldsymbol{c} \in \{0, 1\}^n$ to be a codeword is that it satisfy Equation 7.2–5. Since rows of $\boldsymbol{G}$ are codewords, we conclude that

$$\boldsymbol{G}\boldsymbol{H}^t = \boldsymbol{0} \tag{7.2–6}$$

In the special case of systematic codes, where $\boldsymbol{G} = [\boldsymbol{I}_k \mid \boldsymbol{P}]$, the parity check matrix is given by

$$\boldsymbol{H} = \left[-\boldsymbol{P}^t \mid \boldsymbol{I}_{n-k}\right] \tag{7.2–7}$$

which obviously satisfies $\boldsymbol{G}\boldsymbol{H}^t = \boldsymbol{0}$. For binary codes $-\boldsymbol{P}^t = \boldsymbol{P}^t$ and $\boldsymbol{H} = \left[\boldsymbol{P}^t \mid \boldsymbol{I}_{n-k}\right]$.

EXAMPLE 7.2–1. Consider a (7, 4) linear block code with

$$\boldsymbol{G} = [\boldsymbol{I}_4 \mid \boldsymbol{P}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{7.2–8}$$

Obviously this is a systematic code. The parity check matric for this code is obtained from Equation 7.2–7 as

$$\boldsymbol{H} = \left[\boldsymbol{P}^t \mid \boldsymbol{I}_{n-k}\right] = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{7.2–9}$$

If $\boldsymbol{u} = (u_1, u_2, u_3, u_4)$ is an information sequence, the corresponding codeword $\boldsymbol{c} = (c_1, c_2, \ldots, c_7)$ is given by

$$\begin{aligned} c_1 &= u_1 \\ c_2 &= u_2 \\ c_3 &= u_3 \\ c_4 &= u_4 \\ c_5 &= u_1 + u_2 + u_3 \\ c_6 &= u_2 + u_3 + u_4 \\ c_7 &= u_1 + u_2 + u_4 \end{aligned} \tag{7.2–10}$$

and from Equations 7.2–10 it can be easily verified that all codewords $\boldsymbol{c}$ satisfy Equation 7.2–5.

7.2–2 Weight and Distance for Linear Block Codes

The *weight* of a codeword $\boldsymbol{c} \in \mathcal{C}$ is denoted by $w(\boldsymbol{c})$ and is the number of nonzero components of that codeword. Since $\mathbf{0}$ is a codeword of all linear block codes, we conclude that each linear block code has one codeword of weight zero. The *Hamming distance* between two codewords $\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C}$, denoted by $d(\boldsymbol{c}_1, \boldsymbol{c}_2)$, is the number of components at which $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ differ. It is clear that the weight of a codeword is its distance from $\mathbf{0}$.

The distance between $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ is the weight of $\boldsymbol{c}_1 - \boldsymbol{c}_2$, and since in linear block codes $\boldsymbol{c}_1 - \boldsymbol{c}_2$ is a codeword, then $d(\boldsymbol{c}_1, \boldsymbol{c}_2) = w(\boldsymbol{c}_1 - \boldsymbol{c}_2)$. We clearly see that in linear block codes there exists a one-to-one correspondence between weight and the distance between codewords. This means that the set of possible distances from any codeword $\boldsymbol{c} \in \mathcal{C}$ to all other codewords is equal to the set of weights of different codewords, and thus is independent of $\boldsymbol{c}$. In other words, in a linear block code, looking from any codeword to all other codewords, one observes the same set of distance, regardless of the codeword one is looking from. Also note that in binary linear block codes we can substitute $\boldsymbol{c}_1 - \boldsymbol{c}_2$ with $\boldsymbol{c}_1 + \boldsymbol{c}_2$.

The *minimum distance* of a code is the minimum of all possible distances between distinct codewords of the code, i.e.,

$$d_{\min} = \min_{\substack{\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C} \\ \boldsymbol{c}_1 \neq \boldsymbol{c}_2}} d(\boldsymbol{c}_1, \boldsymbol{c}_2) \tag{7.2–11}$$

The *minimum weight* of a code is the minimum of the weights of all nonzero codewords, which for linear block codes is equal to the minimum distance.

$$w_{\min} = \min_{\substack{\boldsymbol{c} \in \mathcal{C} \\ \boldsymbol{c} \neq \mathbf{0}}} w(\boldsymbol{c}) \tag{7.2–12}$$

There exists a close relation between the minimum weight of a linear block code and the columns of the parity check matrix $\boldsymbol{H}$. We have previously seen that the necessary and sufficient condition for $\boldsymbol{c} \in \{0, 1\}^n$ to be a codeword is that $\boldsymbol{c}\boldsymbol{H}^t = \mathbf{0}$. If we choose $\boldsymbol{c}$ to be a codeword of minimum weight, from this relation we conclude that $w_{\min}$ (or $d_{\min}$) columns of $\boldsymbol{H}$ are linearly dependent. On the other hand, since there exists no codeword of weight less than $d_{\min}$, no fewer than $d_{\min}$ columns of $\boldsymbol{H}$ can be linearly dependent. Therefore, $d_{\min}$ represents the minimum number of columns of $\boldsymbol{H}$ that can be linearly dependent. In other words the column space of $\boldsymbol{H}$ has dimension $d_{\min} - 1$.

In certain modulation schemes there exists a close relation between Hamming distance and Euclidean distance of the codewords. In binary antipodal signaling—for instance, BPSK modulation—the 0 and 1 components of a codeword $\boldsymbol{c} \in \mathcal{C}$ are mapped to $-\sqrt{\mathcal{E}_c}$ and $+\sqrt{\mathcal{E}_c}$, respectively. Therefore if $\boldsymbol{s}$ is the vector corresponding to the modulated sequence of codeword $\boldsymbol{c}$, we have

$$s_{mj} = (2c_{mj} - 1)\sqrt{\mathcal{E}_c}, \qquad 1 \leq j \leq n, \ 1 \leq m \leq M \tag{7.2–13}$$

and therefore,

$$d^2_{\boldsymbol{s}_m, \boldsymbol{s}_{m'}} = 4\mathcal{E}_c d(\boldsymbol{c}_m, \boldsymbol{c}_{m'}) \tag{7.2–14}$$

where $d_{s_m,s_{m'}}$ denotes the Euclidean distance between the modulated sequences and $d(\mathbf{c}_m, \mathbf{c}_{m'})$ is the Hamming distance between the corresponding codewords. From the above we have

$$d^2_{\text{E min}} = 4\mathcal{E}_c d_{\min} \tag{7.2–15}$$

where $d_{\text{E min}}$ is the minimum Euclidean distance of the BPSK modulated sequences corresponding to the codewords. Using Equation 7.1–8, we conclude that

$$d^2_{\text{E min}} = 4R_c\mathcal{E}_b d_{\min} \tag{7.2–16}$$

For the binary orthogonal modulations, e.g., binary orthogonal FSK, we similarly have

$$d^2_{\text{E min}} = 2R_c\mathcal{E}_b d_{\min} \tag{7.2–17}$$

7.2–3 The Weight Distribution Polynomial

An (n, k) code has 2^k codewords that can have weights between 0 and n. In any linear block code there exists one codeword of weight 0, and the weights of nonzero codewords can be between $d_{\min}$ and n. The *weight distribution polynomial* (WEP) or *weight enumeration function* (WEF) of a code is a polynomial that specifies the number of codewords of different weights in a code. The weight distribution polynomial or weight enumeration function is denoted by $A(Z)$ and is defined by

$$A(Z) = \sum_{i=0}^{n} A_i Z^i = 1 + \sum_{i=d_{\min}}^{n} A_i Z^i \tag{7.2–18}$$

where A_i denotes the number of codewords of weight i. The following properties of the weight enumeration function for linear block codes are straightforward:

$$\begin{aligned} A(1) &= \sum_{i=0}^{n} A_i = 2^k \\ A(0) &= 1 \end{aligned} \tag{7.2–19}$$

The weight enumeration function for many block codes is unknown. For low rate codes the weight enumeration function can be obtained by using a computer search. The *MacWilliams identity* expresses the weight enumeration function of a code in terms of the weight enumeration function of its dual code. By this identity, the weight enumeration function of a code $A(Z)$ is related to the weight enumeration function of its dual code $A_d(Z)$ by

$$A(Z) = 2^{-(n-k)}(1 + Z)^n A_d\left(\frac{1 - Z}{1 + Z}\right) \tag{7.2–20}$$

The weight enumeration function of a code is closely related to the distance enumerator function of a constellation as defined in Equation 4.2–74. Note that for a linear block code, the set of distances seen from any codeword to other codewords is independent of the codeword from which these distances are seen. Therefore, in linear block codes the error bound is independent of the transmitted codeword, and thus, without loss of generality, we can always assume that the all-zero codeword **0** is transmitted. The value of d^2 in Equation 4.2–74 depends on the modulation scheme. For BPSK modulation from Equation 7.2–14 we have

$$d_{\mathrm{E}}^2(\boldsymbol{s}_m) = 4\mathcal{E}_b R_c w(\boldsymbol{c}_m) \tag{7.2–21}$$

where $d_{\mathrm{E}}(\boldsymbol{s}_m)$ denotes the Euclidean distance between $\boldsymbol{s}_m$ and the modulated sequence corresponding to **0**. For orthogonal binary FSK modulation we have

$$d_{\mathrm{E}}^2(\boldsymbol{s}_m) = 2\mathcal{E}_b R_c w(\boldsymbol{c}_m) \tag{7.2–22}$$

The distance enumerator function for BPSK is given by

$$T(X) = \sum_{i=d_{\min}}^{n} A_i X^{4R_c\mathcal{E}_b i} = (A(Z) - 1)\big|_{Z=X^{4R_c\mathcal{E}_b}} \tag{7.2–23}$$

and for orthogonal BFSK by

$$T(X) = \sum_{i=d_{\min}}^{n} A_i X^{2R_c\mathcal{E}_b i} = (A(Z) - 1)\big|_{Z=X^{2R_c\mathcal{E}_b}} \tag{7.2–24}$$

Another version of the weight enumeration function provides information about the weight of the codewords as well as the weight of the corresponding information sequences. This polynomial is called the *input-output weight enumeration function* (IOWEF), denoted by $B(Y, Z)$ and is defined as

$$B(Y, Z) = \sum_{i=0}^{n} \sum_{j=0}^{k} B_{ij} Y^j Z^i \tag{7.2–25}$$

where B_{ij} is the number of codewords of weight i that are generated by information sequences of weight j. Clearly,

$$A_i = \sum_{j=0}^{k} B_{ij} \tag{7.2–26}$$

and for linear block codes we have $B(0, 0) = B_{00} = 1$. It is also clear that

$$A(Z) = B(Y, Z)\big|_{Y=1} \tag{7.2–27}$$

A third form of the weight enumeration function, called the *conditional weight enumeration function* (CWEF), is defined by

$$B_j(Z) = \sum_{i=0}^{n} B_{ij} Z^i \tag{7.2–28}$$

and it represents the weight enumeration function of all codewords corresponding to information sequences of weight j. From Equations 7.2–28 and 7.2–25 it is easy to see that

$$B_j(Z) = \frac{1}{j!}\frac{\partial^j}{\partial Y^j} B(Y, Z)\Big|_{Y=0} \tag{7.2–29}$$

EXAMPLE 7.2–2. In the code discussed in Example 7.2–1, there are $2^4 = 16$ codewords with possible weights between 0 and 7. Substituting all possible information sequences of the form $\boldsymbol{u} = (u_1, u_2, u_3, u_4)$ and generating the codewords, we can verify that for this code $d_{\min} = 3$ and there are 7 codewords of weight 3 and 7 codewords of weight 4. There exist one codeword of weight 7 and one codeword of weight 0. Therefore,

$$A(Z) = 1 + 7Z^3 + 7Z^4 + Z^7 \tag{7.2–30}$$

It is also easy to verify that for this code

$$\begin{array}{llll} B_{00} = 1 & B_{31} = 3 & B_{32} = 3 & B_{33} = 1 \\ B_{41} = 1 & B_{42} = 3 & B_{43} = 3 & B_{74} = 1 \end{array}$$

Hence,

$$B(Y, Z) = 1 + 3YZ^3 + 3Y^2Z^3 + Y^3Z^3 + YZ^4 + 3Y^2Z^4 + 3Y^3Z^4 + Y^4Z^7 \tag{7.2–31}$$

and

$$\begin{aligned} B_0(Z) &= 1 \\ B_1(Z) &= 3Z^3 + Z^4 \\ B_2(Z) &= 3Z^3 + 3Z^4 \\ B_3(Z) &= Z^3 + 3Z^4 \\ B_4(Z) &= Z^7 \end{aligned} \tag{7.2–32}$$

7.2–4 Error Probability of Linear Block Codes

Two types of error probability can be studied when linear block codes are employed. The *block error probability* or *word error probability* is defined as the probability of transmitting a codeword $\boldsymbol{c}_m$ and detecting a different codeword $\boldsymbol{c}_{m'}$. The second type of error probability is the *bit error probability*, defined as the probability of receiving a transmitted information bit in error.

Block Error Probability

Linearity of the code guarantees that the distances from $\boldsymbol{c}_m$ to all other codewords are independent of the choice of $\boldsymbol{c}_m$. Therefore, without loss of generality we can assume that the all-zero codeword $\boldsymbol{0}$ is transmitted.

To determine the block (word) error probability P_e, we note that an error occurs if the receiver declares any codeword $\boldsymbol{c}_m \neq \boldsymbol{0}$ as the transmitted codeword. The probability of this event is denoted by the pairwise error probability $P_{\boldsymbol{0}\to\boldsymbol{c}_m}$, as defined in

Section 4.2–3. Therefore,

$$P_e \leq \sum_{\substack{c_m \in C \\ c_m \neq \mathbf{0}}} P_{\mathbf{0} \to c_m} \tag{7.2–33}$$

where in general $P_{\mathbf{0} \to c_m}$ depends on the Hamming distance between $\mathbf{0}$ and $\boldsymbol{c}_m$, which is equal to $w(\boldsymbol{c}_m)$, in a way that depends on the modulation scheme employed for transmission of the codewords. Since for codewords of equal weight we have the same $P_{\mathbf{0} \to c_m}$, we conclude that

$$P_e \leq \sum_{i=d_{\min}}^{n} A_i P_2(i) \tag{7.2–34}$$

where $P_2(i)$ denotes the *pairwise error probability* (PEP) between two codewords with Hamming distance i.

From Equation 6.8–9 we know that

$$P_{\mathbf{0} \to c_m} \leq \prod_{i=1}^{n} \sum_{y_i \in \mathscr{Y}} \sqrt{p(y_i|0) p(y_i|c_{mi})} \tag{7.2–35}$$

Following Example 6.8–1 we define

$$\Delta = \sum_{y \in \mathscr{Y}} \sqrt{p(y|0) p(y|1)} \tag{7.2–36}$$

With this definition, Equation 7.2–35 reduces to

$$P_{\mathbf{0} \to c_m} = P_2(w(\boldsymbol{c}_m)) \leq \Delta^{w(c_m)} \tag{7.2–37}$$

Substituting this result into Equation 7.2–34 results in

$$P_e \leq \sum_{i=d_{\min}}^{n} A_i \Delta^i \tag{7.2–38}$$

or

$$P_e \leq A(\Delta) - 1 \tag{7.2–39}$$

where $A(Z)$ is the weight enumerating function of the linear block code.

From the inequality

$$\sum_{y \in \mathscr{Y}} \left(\sqrt{p(y|0)} - \sqrt{p(y|1)} \right)^2 \geq 0 \tag{7.2–40}$$

we easily conclude that

$$\Delta = \sum_{y \in \mathscr{Y}} \sqrt{p(y|0) p(y|1)} \leq 1 \tag{7.2–41}$$

and hence, for $i \geq d_{\min}$,

$$\Delta^i \leq \Delta^{d_{\min}} \tag{7.2–42}$$

Using this result in Equation 7.2–38 yields the simpler, but looser, bound

$$P_e \le (2^k - 1)\Delta^{d_{\min}} \tag{7.2–43}$$

Bit Error Probability

In general, errors at different locations of an information sequence of length k can occur with different probabilities. We define the average of these error probabilities as the bit error probability for a linear block code. We again assume that the all-zero sequence is transmitted; then the probability that a specific codeword of weight i will be decoded at the detector is equal to $P_2(i)$. The number of codewords of weight i that correspond to information sequences of weight j is denoted by B_{ij}. Therefore, when $\mathbf{0}$ is transmitted, the expected number of information bits received in error is given by

$$\bar{b} \le \sum_{j=0}^{k} j \sum_{i=d_{\min}}^{n} B_{ij} P_2(i) \tag{7.2–44}$$

Since for $0 < i < d_{\min}$ we have $B_{ij} = 0$, we can write this as

$$\bar{b} \le \sum_{j=0}^{k} j \sum_{i=0}^{n} B_{ij} P_2(i) \tag{7.2–45}$$

The (average) bit error probability of the linear block code P_b is defined as the ratio of the expected number of bits received in error to the total number of transmitted bits, i.e.,

$$\begin{aligned} P_b &= \frac{\bar{b}}{k} \\ &\le \frac{1}{k} \sum_{j=0}^{k} j \sum_{i=0}^{n} B_{ij} P_2(i) \\ &\le \frac{1}{k} \sum_{j=0}^{k} j \sum_{i=0}^{n} B_{ij} \Delta^{i} \end{aligned} \tag{7.2–46}$$

where in the last step we have used Equation 7.2–37. From Equation 7.2–28 we see that the last sum is simply $B_j(\Delta)$; therefore,

$$P_b \le \frac{1}{k} \sum_{j=0}^{k} j B_j(\Delta) \tag{7.2–47}$$

We can also express the bit error probability in terms of the IOWEF by using Equation 7.2–25 as

$$
\begin{aligned}
P_b &\leq \frac{1}{k} \sum_{i=0}^{n} \sum_{j=0}^{k} j B_{ij} \Delta^i \\
&= \frac{1}{k} \frac{\partial}{\partial Y} B(Y, Z)\Big|_{Y=1, Z=\Delta}
\end{aligned}
\tag{7.2–48}
$$

7.3
SOME SPECIFIC LINEAR BLOCK CODES

In this section, we briefly describe some linear block codes that are frequently encountered in practice and list their important parameters. Additional classes of linear codes are introduced in our study of cyclic codes in Section 7.9.

7.3–1 Repetition Codes

A binary repetition code is an $(n, 1)$ code with two codewords of length n. One codeword is the all-zero codeword, and the other one is the all-one codeword. This code has a rate of $R_c = \frac{1}{n}$ and a minimum distance of $d_{\min} = n$. The dual of a repetition code is an $(n, n-1)$ code consisting of all binary sequences of length n with even parity. The minimum distance of the dual code is clearly $d_{\min} = 2$.

7.3–2 Hamming Codes

Hamming codes are one of the earliest codes studied in coding theory. Hamming codes are linear block codes with parameters $n = 2^m - 1$ and $k = 2^m - m - 1$, for $m \geq 3$. Hamming codes are best described in terms of their parity check matrix $\boldsymbol{H}$ which is an $(n-k) \times n = m \times (2^m - 1)$ matrix. The $2^m - 1$ columns of $\boldsymbol{H}$ consist of all possible binary vectors of length m excluding the all-zero vector. The rate of a Hamming code is given by

$$
R_c = \frac{2^m - m - 1}{2^m - 1} \tag{7.3–1}
$$

which is close to 1 for large values of m.

Since the columns of $\boldsymbol{H}$ include all nonzero sequences of length m, the sum of any two columns is another column. In other words, there always exist three columns that are linearly dependent. Therefore, for Hamming codes, independent of the value of m, $d_{\min} = 3$.

The weight distribution polynomial for the class of Hamming (n, k) codes is known and is expressed as (see Problem 7.23)

$$A(Z) = \frac{1}{n+1}\left[(1+Z)^n + n(1+Z)^{(n-1)/2}(1-Z)^{(n+1)/2}\right] \tag{7.3–2}$$

EXAMPLE 7.3–1. To generate the $\boldsymbol{H}$ matrix for a (7, 4) Hamming code (corresponding to $m = 3$), we have to use all nonzero sequences of length 3 as columns of $\boldsymbol{H}$. We can arrange these columns in such a way that the resulting code is systematic as

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{7.3–3}$$

This is the parity check matrix derived in Example 7.2–1 and given by Equation 7.2–9.

7.3–3 Maximum-Length Codes

Maximum-length codes are duals of Hamming codes; therefore these are a family of $(2^m - 1, m)$ codes for $m \geq 3$. The generator matrix of a maximum-length code is the parity check matrix of a Hamming code, and therefore its columns are all sequences of length m with the exception of the all-zero sequence. In Problem 7.23 it is shown that maximum-length codes are constant-weight codes; i.e., all codewords, except the all-zero codeword, have the same weight, and this weight is equal to 2^{m-1}. Therefore, the weight enumeration function for these codes is given by

$$A(Z) = 1 + (2^m - 1)Z^{m-1} \tag{7.3–4}$$

Using this weight distribution function and applying the MacWilliams identity given in Equation 7.2–20, we can derive the weight enumeration function of the Hamming code as given in Equation 7.3–2.

7.3–4 Reed-Muller Codes

Reed-Muller codes introduced by Reed (1954) and Muller (1954) are a class of linear block codes with flexible parameters that are particularly interesting due to the existence of simple decoding algorithms for them.

A Reed-Muller code with block length $n = 2^m$ and *order* $r < m$ is an (n, k) linear block code with

$$\begin{aligned} n &= 2^m \\ k &= \sum_{i=0}^{r} \binom{m}{i} \\ d_{\min} &= 2^{m-r} \end{aligned} \tag{7.3–5}$$

whose generator matrix is given by

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_0 \\ \boldsymbol{G}_1 \\ \boldsymbol{G}_2 \\ \vdots \\ \boldsymbol{G}_r \end{bmatrix} \tag{7.3–6}$$

where $\boldsymbol{G}_0$ is a $1 \times n$ matrix of all 1s

$$\boldsymbol{G}_0 = [1 \quad 1 \quad 1 \quad \ldots \quad 1] \tag{7.3–7}$$

and $\boldsymbol{G}_1$ is an $m \times n$ matrix whose columns are distinct binary sequences of length m put in natural binary order.

$$\boldsymbol{G}_1 = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 1 \\ 0 & 0 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & \cdots & 1 & 1 \\ 0 & 1 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{7.3–8}$$

$\boldsymbol{G}_2$ is an $\binom{m}{2} \times n$ matrix whose rows are obtained by bitwise multiplication of two rows of $\boldsymbol{G}_2$ at a time. Similarly, $\boldsymbol{G}_i$ for $2 < i \leq r$ is a $\binom{m}{r} \times n$ matrix whose rows are obtained by bitwise multiplication of r rows of $\boldsymbol{G}_2$ at a time.

EXAMPLE 7.3–2. The first-order Reed-Muller code with block length 8 is an $(8, 4)$ code with generator matrix

$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{7.3–9}$$

This code can be obtained from a $(7, 3)$ maximum-length code by adding one extra parity bit to make the overall weight of each codeword even. This code has a minimum distance of 4. The second-order Reed-Muller code with block length 8 has the generator matrix

$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7.3–10}$$

and has a minimum distance of 2.

7.3–5 Hadamard Codes

Hadamard signals were introduced in Section 3.2–4 as examples of orthogonal signaling schemes. A Hadamard code is obtained by selecting as codewords the rows of a Hadamard matrix. A Hadamard matrix $\boldsymbol{M}_n$ is an $n \times n$ matrix (n is an even integer) of 1s and 0s with the property that any row differs from any other row in exactly $\frac{n}{2}$ positions.† One row of the matrix contains all zeros. The other rows each contain $\frac{n}{2}$ zeros and $\frac{n}{2}$ ones.

For $n = 2$, the Hadamard matrix is

$$\boldsymbol{M}_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{7.3–11}$$

Furthermore, from $\boldsymbol{M}_n$, we can generate the Hadamard matrix $\boldsymbol{M}_{2n}$ according to the relation

$$\boldsymbol{M}_{2n} = \begin{bmatrix} \boldsymbol{M}_n & \boldsymbol{M}_n \\ \boldsymbol{M}_n & \overline{\boldsymbol{M}}_n \end{bmatrix} \tag{7.3–12}$$

where $\overline{\boldsymbol{M}}_n$ denotes the complement (0s replaced by 1s and vice versa) of $\boldsymbol{M}_n$. Thus, by substituting Equation 7.3–11 into Equation 7.3–12, we obtain

$$\boldsymbol{M}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{7.3–13}$$

The complement of M4 is

$$\overline{\boldsymbol{M}}_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{7.3–14}$$

Now the rows of $\boldsymbol{M}_4$ and $\overline{\boldsymbol{M}}_4$ form a linear binary code of block length $n = 4$ having $2n = 8$ codewords. The minimum distance of the code is $d_{\text{min}} = \frac{n}{2} = 2$.

By repeated application of Equation 7.3–12, we can generate Hadamard codes with block length $n = 2^m$, $k = \log_2 2n = \log_2 2^{m+1} = m + 1$, and $d_{\text{min}} = \frac{n}{2} = 2^{m-1}$, where m is a positive integer. In addition to the important special cases where $n = 2^m$, Hadamard codes of other block lengths are possible, but the resulting codes are not linear.

---

†In Section 3.2–4 the elements of the Hadamard matrix were denoted $+1$ and -1, resulting in mutually orthogonal rows. We also note that the $M = 2^k$ signal waveforms, constructed from Hadamard codewords by mapping each bit in a codeword into a binary PSK signal, are orthogonal.

7.3–6 Golay Codes

The *Golay code* (Golay (1949)) is a binary linear (23, 12) code with $d_{\min} = 7$. The *extended Golay code* is obtained by adding an overall parity bit to the (23, 12) Golay code such that each codeword has even parity. The resulting code is a binary linear (24, 12) code with $d_{\min} = 8$. The weight distribution polynomials of Golay code and extended Golay code are known and are given by

$$\begin{aligned} A_{\mathrm{G}}(Z) &= 1 + 253Z^7 + 506Z^8 + 1288Z^{11} + 1288Z^{12} + 506Z^{15} + 253Z^{16} + Z^{23} \\ A_{\mathrm{EG}}(Z) &= 1 + 759Z^8 + 2576Z^{12} + 759Z^{16} + Z^{24} \end{aligned} \tag{7.3–15}$$

We discuss the generation of the Golay code in Section 7.9–5.

7.4

OPTIMUM SOFT DECISION DECODING OF LINEAR BLOCK CODES

In this section, we derive the performance of linear binary block codes on an AWGN channel when optimum (unquantized) *soft decision decoding* is employed at the receiver. The bits of a codeword may be transmitted by any one of the binary signaling methods described in Chapter 3. For our purposes, we consider binary (or quaternary) coherent PSK, which is the most efficient method, and binary orthogonal FSK with either coherent detection or noncoherent detection.

From Chapter 4, we know that the optimum receiver, in the sense of minimizing the average probability of a codeword error, for the AWGN channel can be realized as a parallel bank of $M = 2^k$ filters matched to the M possible transmitted waveforms. The outputs of the M matched filters at the end of each signaling interval, which encompasses the transmission of n binary symbols in the codeword, are compared, and the codeword corresponding to the largest matched filter output is selected. Alternatively, M cross-correlators can be employed. In either case, the receiver implementation can be simplified. That is, an equivalent optimum receiver can be realized by use of a single filter (or cross-correlator) matched to the binary PSK waveform used to transmit each bit in the codeword, followed by a decoder that forms the M decision variables corresponding to the M codewords.

To be specific, let $r_j,\ j = 1, 2, \ldots, n$, represent the n sampled outputs of the matched filter for any particular codeword. Since the signaling is binary coherent PSK, the output r_j may be expressed either as

$$r_j = \sqrt{\mathcal{E}_c} + n_j \tag{7.4–1}$$

when the jth bit of a codeword is a 1, or as

$$r_j = -\sqrt{\mathcal{E}_c} + n_j \tag{7.4–2}$$

when the jth bit is a 0. The variables $\{n_j\}$ represent additive white Gaussian noise at the sampling instants. Each n_j has zero mean and variance $\frac{1}{2}N_0$. From knowledge of the

M possible transmitted codewords and upon reception of $\{r_j\}$, the optimum decoder forms the M correlation metrics

$$CM_m = C(\boldsymbol{r}, \boldsymbol{c}_m) = \sum_{j=1}^{n} (2c_{mj} - 1) r_j, \qquad m = 1, 2, \ldots, M \tag{7.4–3}$$

where c_{mj} denotes the bit in the jth position of the mth codeword. Thus, if $c_{mj} = 1$, the weighting factor $2c_{mj} - 1 = 1$; and if $c_{mj} = 0$, the weighting factor $2c_{mj} - 1 = -1$. In this manner, the weighting $2c_{mj} - 1$ aligns the signal components in $\{r_j\}$ such that the correlation metric corresponding to the actual transmitted codeword will have a mean value $n\sqrt{\mathcal{E}_c}$, while the other $M - 1$ metrics will have smaller mean values.

Although the computations involved in forming the correlation metrics for soft decision decoding according to Equation 7.4–3 are relatively simple, it may still be impractical to compute Equation 7.4–3 for all the possible codewords when the number of codewords is large, e.g., $M > 2^{10}$. In such a case it is still possible to implement soft decision decoding using algorithms which employ techniques for discarding improbable codewords without computing their entire correlation metrics as given by Equation 7.4–3. Several different types of soft decision decoding algorithms have been described in the technical literature. The interested reader is referred to the papers by Forney (1966b), Weldon (1971), Chase (1972), Wainberg and Wolf (1973), Wolf (1978), and Matis and Modestino (1982).

Block and Bit Error Probability in Soft Decision Decoding

We can use the general bounds on the block error probability derived in Equations 7.2–39 and 7.2–43 to find bounds on the block error probability for soft decision decoding. The value of Δ defined by Equation 7.2–36 has to be found under the specific modulation employed to transmit codeword components. In Example 6.8–1 it was shown that for BPSK modulation we have $\Delta = e^{-\mathcal{E}_c/N_0}$, and since $\mathcal{E}_c = R_c\mathcal{E}_b$, we obtain

$$P_e \leq (A(Z) - 1)\Big|_{Z=e^{-\frac{R_c\mathcal{E}_b}{N_0}}} \tag{7.4–4}$$

where $A(Z)$ is the weight enumerating polynomial of the code.

The simple bound of Equation 7.2–43 under soft decision decoding reduces to

$$P_e \leq (2^k - 1)e^{-R_c d_{\min}\mathcal{E}_b/N_0} \tag{7.4–5}$$

In Problem 7.18 it is shown that for binary orthogonal signaling, for instance, orthogonal BFSK, we have $\Delta = e^{-\mathcal{E}_c/2N_0}$. Using this result, we obtain the simple bound

$$P_e \leq (2^k - 1)e^{-R_c d_{\min}\mathcal{E}_b/2N_0} \tag{7.4–6}$$

for orthogonal BFSK modulation.

Using the inequality $2^k - 1 < 2^k = e^{k\ln 2}$, we obtain

$$P_e \leq e^{-\gamma_b\left(R_c d_{\min} - \frac{k\ln 2}{\gamma_b}\right)} \qquad \text{for BPSK} \tag{7.4–7}$$

and

$$P_e \le e^{-\frac{\gamma_b}{2}\left(R_c d_{\min} - \frac{k \ln 2}{\gamma_b}\right)} \qquad \text{for orthogonal BFSK} \tag{7.4–8}$$

where as usual γ_b denotes $\mathcal{E}_b/N_0$, the SNR per bit.

When the upper bound in Equation 7.4–7 is compared with the performance of an uncoded binary PSK system, which is upper-bounded as $\frac{1}{2}\exp(-\gamma_b)$, we find that coding yields a gain of approximately $10\log(R_c d_{\min} - k\ln 2/\gamma_b)$ dB. We may call this the *coding gain*. We note that its value depends on the code parameters and also on the SNR per bit γ_b. For large values of γ_b, the limit of the coding gain, i.e., $R_c d_{\min}$, is called the *asymptotic coding gain*.

Similar to the block error probability, we can use Equation 7.2–48 to bound the bit error probability for BFSK and orthogonal BFSK modulation. We obtain

$$\begin{aligned} P_b &\le \frac{1}{k}\frac{\partial}{\partial Y}B(Y, Z)\Big|_{Y=1, Z=\exp\left(-\frac{R_c\mathcal{E}_b}{N_0}\right)} && \text{for BPSK} \\ P_b &\le \frac{1}{k}\frac{\partial}{\partial Y}B(Y, Z)\Big|_{Y=1, Z=\exp\left(-\frac{R_c\mathcal{E}_b}{2N_0}\right)} && \text{for orthogonal BFSK} \end{aligned} \tag{7.4–9}$$

Soft Decision Decoding with Noncoherent Detection

In noncoherent detection of binary orthogonal FSK signaling, the performance is further degraded by the noncoherent combining loss. Here the input variables to the decoder are

$$\begin{cases} r_{0j} = |\sqrt{\mathcal{E}_c} + N_{0j}|^2 \\ r_{0j} = |N_{1j}|^2 \end{cases} \tag{7.4–10}$$

for $j = 1, 2, \ldots, n$, where $\{N_{0j}\}$ and $\{N_{1j}\}$ represent complex-valued mutually statistically independent Gaussian random variables with zero mean and variance $2N_0$. The correlation metric CM_1 is given as

$$CM_1 = \sum_{j=1}^{n} r_{0j} \tag{7.4–11}$$

while the correlation metric corresponding to the codeword having weight w_m is statistically equivalent to the correlation metric of a codeword in which $c_{mj} = 1$ for $1 \le j \le w_m$ and $c_{mj} = 0$ for $w_m + 1 \le j \le n$. Hence, CM_m may be expressed as

$$CM_m = \sum_{j=1}^{w_m} r_{1j} + \sum_{j=w_m+1}^{n} r_{0j} \tag{7.4–12}$$

The difference between CM_1 and CM_m is

$$CM_1 - CM_m = \sum_{j=1}^{w_m}(r_{0j} - r_{1j}) \tag{7.4–13}$$

and the pairwise error probability (PEP) is simply the probability that $CM_1 - CM_m < 0$. But this difference is a special case of the general quadratic form in complex-valued Gaussian random variables considered in Chapter 11 and in Appendix B. The expression for the probability of error in deciding between CM_1 and CM_m is (see Section 11.1–1)

$$P_2(m) = \frac{1}{2^{2w_m-1}} \exp\left(-\frac{1}{2}\gamma_b R_c w_m\right) \sum_{i=0}^{w_m-1} K_i \left(\frac{1}{2}\gamma_b R_c w_m\right)^i \tag{7.4–14}$$

where, by definition,

$$K_i = \frac{1}{i!} \sum_{r=0}^{w_m-1-i} \binom{2w_m-1}{r} \tag{7.4–15}$$

The union bound obtained by summing $P_2(m)$ over $2 \leq m \leq M$ provides us with an upper bound on the probability of a codeword error.

As an alternative, we may use the minimum distance instead of the weight distribution to obtain the looser upper bound

$$P_e \leq \frac{M-1}{2^{2d_{\min}-1}} \exp\left(-\frac{1}{2}\gamma_b R_c d_{\min}\right) \sum_{i=0}^{d_{\min}-1} K_i \left(\frac{1}{2}\gamma_b R_c d_{\min}\right)^i \tag{7.4–16}$$

A measure of the noncoherent combining loss inherent in the square-law detection and combining of the n elementary binary FSK waveforms in a codeword can be obtained from Figure 11.1–1, where $d_{\min}$ is used in place of L. The loss obtained is relative to the case in which the n elementary binary FSK waveforms are first detected coherently and combined, and then the sums are square-law-detected or envelope-detected to yield the M decision variables. The binary error probability for the latter case is

$$P_2(m) = \frac{1}{2} \exp\left(-\frac{1}{2}\gamma_b R_c w_m\right) \tag{7.4–17}$$

and hence

$$P_e \leq \sum_{m=2}^{M} P_2(m) \tag{7.4–18}$$

If $d_{\min}$ is used instead of the weight distribution, the union bound for the codeword error probability in the latter case is

$$P_e \leq \frac{1}{2}(M-1) \exp\left(-\frac{1}{2}\gamma_b R_c d_{\min}\right) \tag{7.4–19}$$

similar to Equation 7.4–8.

We have previously seen in Equation 7.1–10 that the channel bandwidth required to transmit the coded waveforms, when binary PSK is used to transmit each bit, is given by

$$W = \frac{R}{R_c} \tag{7.4–20}$$

From Equation 4.6–7, the bandwidth requirement for an uncoded BPSK scheme is R. Therefore, the *bandwidth expansion factor* B_e for the coded waveforms is

$$B_e = \frac{1}{R_c} \tag{7.4–21}$$

Comparison with Orthogonal Signaling

We are now in a position to compare the performance characteristics and bandwidth requirements of coded signaling with orthogonal signaling. As we have seen in Chapter 4, orthogonal signals are more power-efficient compared to BPSK signaling, but using them requires large bandwidth. We have also seen that using coded BPSK signals results in a moderate expansion in bandwidth and, at the same time, by providing the coding gain, improves the power efficiency of the system.

Let us consider two systems, one employing orthogonal signaling and one employing coded BPSK signals to achieve the same performance. We use the bounds given in Equations 4.4–17 and 7.4–7 to compare the error probabilities of orthogonal and coded BPSK signals, respectively. To have equal bounds on the error probability, we must have $k = 2R_c d_{\min}$. Under this condition, the dimensionality of the orthogonal signals, given by $N = M = 2^k$, is given by $N = 2^{R_c d_{\min}}$. The dimensionality of the BPSK code waveform is $n = k/R_c = 2d_{\min}$. Since dimensionality is proportional to the bandwidth, we conclude that

$$\frac{W_{\text{orthogonal}}}{W_{\text{coded BPSK}}} = \frac{2^{2R_c d_{\min}}}{2d_{\min}} \tag{7.4–22}$$

For example, suppose we use a (63, 30) binary code that has a minimum distance $d_{\min} = 13$. The bandwidth ratio for orthogonal signaling relative to this code, given by Equation 7.4–22, is roughly 205. In other words, an orthogonal signaling scheme that performs similar to the (63, 30) code requires 205 times the bandwidth of the coded system. This example clearly shows the bandwidth efficiency of coded systems.

7.5
HARD DECISION DECODING OF LINEAR BLOCK CODES

The bounds given in Section 7.4 on the performance of coded signaling waveforms on the AWGN channel are based on the premise that the samples from the matched filter or cross-correlator are *not* quantized. Although this processing yields the best performance, the basic limitation is the computational burden of forming M correlation metrics and comparing these to obtain the largest. The amount of computation becomes excessive when the number M of codewords is large.

To reduce the computational burden, the analog samples can be quantized and the decoding operations are then performed digitally. In this section, we consider the extreme situation in which each sample corresponding to a single bit of a codeword is quantized to two levels: 0 and 1. That is, a *hard decision* is made as to whether each transmitted bit in a codeword is a 0 or a 1. The resulting discrete-time channel (consisting of the modulator, the AWGN channel, and the modulator/demodulator) constitutes a BSC with crossover probability p. If coherent PSK is employed in transmitting and receiving the bits in each codeword, then

$$p = Q\left(\sqrt{\frac{2\mathcal{E}_c}{N_0}}\right) = Q\left(\sqrt{2\gamma_b R_c}\right) \tag{7.5–1}$$

On the other hand, if FSK is used to transmit the bits in each codeword, then

$$p = Q\left(\sqrt{\gamma_b R_c}\right) \tag{7.5–2}$$

for coherent detection and

$$p = \frac{1}{2}\exp\left(-\frac{1}{2}\gamma_b R_c\right) \tag{7.5–3}$$

for noncoherent detection.

Minimum-Distance (Maximum-Likelihood) Decoding

The n bits from the detector corresponding to a received codeword are passed to the decoder, which compares the received codeword with the M possible transmitted codewords and decides in favor of the codeword that is closest in Hamming distance (number of bit positions in which two codewords differ) to the received codeword. This minimum-distance decoding rule is optimum in the sense that it results in a minimum probability of a codeword error for the binary symmetric channel.

A conceptually simple, albeit computationally inefficient, method for decoding is to first add (modulo-2) the received codeword vector to all the M possible transmitted codewords $\boldsymbol{c}_m$ to obtain the error vectors $\boldsymbol{e}_m$. Hence, $\boldsymbol{e}_m$ represents the error event that must have occurred on the channel in order to transform the codeword $\boldsymbol{c}_m$ to the particular received codeword. The number of errors in transforming $\boldsymbol{c}_m$ into the received codeword is just equal to the number of 1s in $\boldsymbol{e}_m$. Thus, if we simply compute the weight of each of the M error vectors $\{\boldsymbol{e}_m\}$ and decide in favor of the codeword that results in the smallest weight error vector, we have, in effect, a realization of the minimum-distance decoding rule.

Syndrome and Standard Array

A more efficient method for hard decision decoding makes use of the parity check matrix $\boldsymbol{H}$. To elaborate, suppose that $\boldsymbol{c}_m$ is the transmitted codeword and $\boldsymbol{y}$ is the received sequence at the output of the detector. In general, $\boldsymbol{y}$ may be expressed as

$$\boldsymbol{y} = \boldsymbol{c}_m + \boldsymbol{e}$$

where $\boldsymbol{e}$ denotes an arbitrary binary error vector. The product $\boldsymbol{y}\boldsymbol{H}^t$ yields

$$\begin{aligned} \boldsymbol{s} &= \boldsymbol{y}\boldsymbol{H}^t \\ &= \boldsymbol{c}_m\boldsymbol{H}^t + \boldsymbol{e}\boldsymbol{H}^t \\ &= \boldsymbol{e}\boldsymbol{H}^t \end{aligned} \tag{7.5–4}$$

where the $(n-k)$-dimensional vector $\boldsymbol{s}$ is called the *syndrome* of the error pattern. In other words, the vector $\boldsymbol{s}$ has components that are zero for all parity check equations that are satisfied and nonzero for all parity check equations that are not satisfied. Thus, $\boldsymbol{s}$ contains the pattern of failures in the parity checks.

We emphasize that the syndrome $\boldsymbol{s}$ is a characteristic of the error pattern and not of the transmitted codeword. If a syndrome is equal to zero, then the error pattern is equal to one of the codewords. In this case we have an *undetected error*. Therefore, an error pattern remains undetected if it is equal to one of the nonzero codewords. Hence, from the 2^n-1 error patterns (the all-zero sequence does not count as an error), 2^k-1 are not detectable; the remaining 2^n-2^k nonzero error patterns can be detected, but not all can be corrected because there are only 2^{n-k} syndromes and, consequently, different error patterns result in the same syndrome. For ML decoding we are looking for the error pattern of least weight among all possible error patterns.

Suppose we construct a decoding table in which we list all the 2^k possible codewords in the first row, beginning with the all-zero codeword $\boldsymbol{c}_1=\boldsymbol{0}$ in the first (leftmost) column. This all-zero codeword also represents the all-zero error pattern. After completing the first row, we put a sequence of length n which has not been included in the first row (i.e., is not a codeword) and among all such sequences has the minimum weight in the first column of the second row, and we call it $\boldsymbol{e}_2$. We complete the second row of the table by adding $\boldsymbol{e}_2$ to all codewords and putting the result in the column corresponding to that codeword. After the second row is complete, we look among all sequences of length n that have not been included in the first two rows and choose a sequence of minimum weight, call it $\boldsymbol{e}_3$, and put it in the first column of the third row; and complete the third row similar to the way we completed the second row. This process is continued until all sequences of length n are used in the table. We obtain an $n\times(n-k)$ table as follows:

$$\begin{array}{ccccc}
\boldsymbol{c}_1=\boldsymbol{0} & \boldsymbol{c}_2 & \boldsymbol{c}_3 & \cdots & \boldsymbol{c}_{2^k} \\
\boldsymbol{e}_2 & \boldsymbol{c}_2+\boldsymbol{e}_2 & \boldsymbol{c}_3+\boldsymbol{e}_2 & \cdots & \boldsymbol{c}_{2^k}+\boldsymbol{e}_2 \\
\boldsymbol{e}_3 & \boldsymbol{c}_2+\boldsymbol{e}_3 & \boldsymbol{c}_3+\boldsymbol{e}_3 & \cdots & \boldsymbol{c}_{2^k}+\boldsymbol{e}_3 \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\boldsymbol{e}_{2^{n-k}} & \boldsymbol{c}_2+\boldsymbol{e}_{2^{n-k}} & \boldsymbol{c}_3+\boldsymbol{e}_{2^{n-k}} & \cdots & \boldsymbol{c}_{2^k}+\boldsymbol{e}_{2^{n-k}}
\end{array}$$

This table is called a *standard array*. Each row, including the first, consists of k possible received sequences that would result from the corresponding error pattern in the first column. Each row is called a *coset*, and the first (leftmost) codeword (or error pattern) is called a *coset leader*. Therefore, a coset consists of all the possible received sequences resulting from a particular error pattern (coset leader). Also note that by construction the coset leader has the lowest weight among all coset members.

EXAMPLE 7.5–1. Let us construct the standard array for the $(5,2)$ systematic code with generator matrix given by

$$\boldsymbol{G}=\begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

TABLE 7.5–1
The Standard Array for Example 7.5–1

00000	**01011**	**10101**	**11110**
00001	01010	10100	11111
00010	01001	10111	11100
00100	01111	10001	11010
01000	00011	11101	10110
10000	11011	00101	01110
11000	10011	01101	00110
10010	11001	00111	01100

This code has a minimum distance $d_{\min} = 3$. The standard array is given in Table 7.5–1. Note that in this code, the coset leaders consist of the all-zero error pattern, five error patterns of weight 1, and two error patterns of weight 2. Although many more double error patterns exist, there is room for only two to complete the table.

Now, suppose that $\boldsymbol{e}_i$ is a coset leader and that $\boldsymbol{c}_m$ was the transmitted codeword. Then the error pattern $\boldsymbol{e}_i$ would result in the received sequence

$$\boldsymbol{y} = \boldsymbol{c}_m + \boldsymbol{e}_i$$

The syndrome is

$$\boldsymbol{s} = \boldsymbol{y}\boldsymbol{H}^t = (\boldsymbol{c}_m + \boldsymbol{e}_i)\boldsymbol{H}^t = \boldsymbol{c}_m\boldsymbol{H}^t + \boldsymbol{e}_i\boldsymbol{H}^t = \boldsymbol{e}_i\boldsymbol{H}^t$$

Clearly, all received sequences in the same coset have the same syndrome, since the latter depends only on the error pattern. Furthermore, each coset has a different syndrome. This means that there exists a one-to-one correspondence between cosets (or coset leaders) and syndromes.

The process of decoding the received sequence $\boldsymbol{y}$ basically involves finding the error sequence of the lowest weight $\boldsymbol{e}_i$ such that $\boldsymbol{s} = \boldsymbol{y}\boldsymbol{H}^t = \boldsymbol{e}_i\boldsymbol{H}^t$. Since each syndrome $\boldsymbol{s}$ corresponds to a single coset, the error sequence $\boldsymbol{e}_i$ is simply the lowest member of the coset, i.e., the coset leader. Therefore, after the syndrome is found, it is sufficient to find the coset leader corresponding to the syndrome and add the coset leader to $\boldsymbol{y}$ to obtain the most likely transmitted codeword.

The above discussion makes it clear that coset leaders are the only error patterns that are correctable. To sum up the above discussion, from all possible $2^n - 1$ nonzero error patterns, $2^k - 1$ corresponding to nonzero codewords are not detectable, and $2^n - 2^k$ are detectable of which only $2^{n-k} - 1$ are correctable.

EXAMPLE 7.5–2. Consider the (5, 2) code with the standard array given in Table 7.5–1. The syndromes versus the most likely error patterns are given in Table 7.5–2.

Now suppose the actual error vector on the channel is

$$\boldsymbol{e} = (1 \quad 0 \quad 1 \quad 0 \quad 0)$$

The syndrome computed for the error is $\boldsymbol{s} = (0 \quad 0 \quad 1)$. Hence, the error determined from the table is $\hat{\boldsymbol{e}} = (0 \quad 0 \quad 0 \quad 0 \quad 1)$. When $\hat{\boldsymbol{e}}$ is added to $\boldsymbol{y}$, the result is a decoding

TABLE 7.5–2
Syndromes and Coset Leaders for Example 7.5–2

Syndrome	Error Pattern
000	00000
001	00001
010	00010
100	00100
011	01000
101	10000
110	11000
111	10010

error. In other words, the (5, 2) code corrects all single errors and only two double errors, namely, (1 1 0 0 0) and (1 0 0 1 0).

7.5–1 Error Detection and Error Correction Capability of Block Codes

It is clear from the discussion above that when the syndrome consists of all zeros, the received codeword is one of the 2^k possible transmitted codewords. Since the minimum separation between a pair of codewords is $d_{\min}$, it is possible for an error pattern of weight $d_{\min}$ to transform one of these 2^k codewords in the code to another codeword. When this happens, we have an *undetected error*. On the other hand, if the actual number of errors is less than $d_{\min}$, the syndrome will have a nonzero weight. When this occurs, we have detected the presence of one or more errors on the channel. Clearly, the (n, k) block code is capable of *detecting* up to $d_{\min} - 1$ errors. Error detection may be used in conjunction with an *automatic repeat-request* (ARQ) scheme for retransmission of the codeword.

The *error correction capability* of a code also depends on the minimum distance. However, the number of correctable error patterns is limited by the number of possible syndromes or coset leaders in the standard array. To determine the error correction capability of an (n, k) code, it is convenient to view the 2^k codewords as points in an n-dimensional space. If each codeword is viewed as the center of a sphere of radius (Hamming distance) t, the largest value that t may have without intersection (or tangency) of any pair of the 2^k spheres is $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$, where $\lfloor x \rfloor$ denotes the largest integer contained in x. Within each sphere lie all the possible received codewords of distance less than or equal to t from the valid codeword. Consequently, any received code vector that falls within a sphere is decoded into the valid codeword at the center of the sphere. This implies that an (n, k) code with minimum distance $d_{\min}$ is capable of correcting $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ errors. Figure 7.5–1 is a two-dimensional representation of the codewords and the spheres.

As described above, a code may be used to detect $d_{\min} - 1$ errors or to correct $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ errors. Clearly, to correct t errors implies that we have detected t errors. However, it is also possible to detect more than t errors if we compromise in the error correction capability of the code. For example, a code with $d_{\min} = 7$ can correct

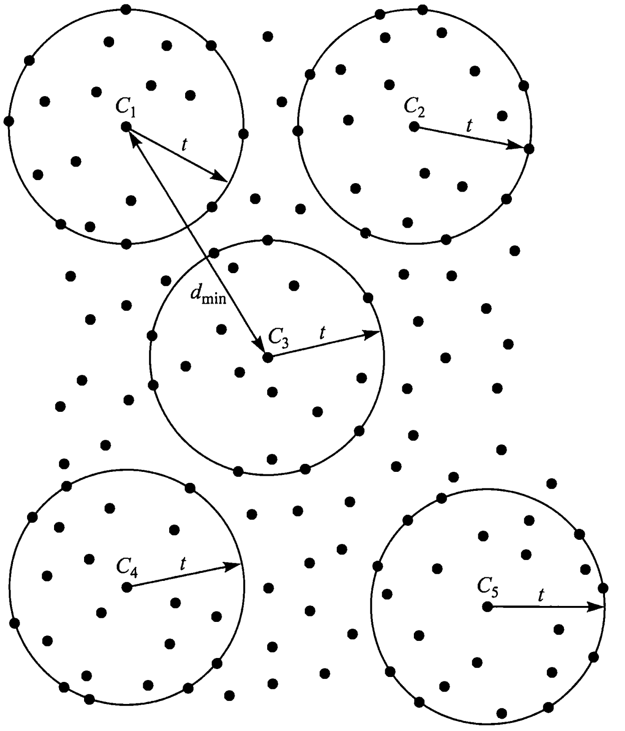


FIGURE 7.5–1
A representation of codewords as center of spheres with radius $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$.

up to $t = 3$ errors. If we wish to detect four errors, we can do so by reducing the radius of the sphere around each codeword from 3 to 2. Thus, patterns with four errors are detectable, but only patterns of two errors are correctable. In other words, when only two errors occur, these are corrected; and when three or four errors occur, the receiver may ask for a retransmission. If more than four errors occur, they will go undetected if the codeword falls within a sphere of radius 2. Similarly, for $d_{\min} = 7$, five errors can be detected and one error corrected. In general, a code with minimum distance $d_{\min}$ can detect e_d errors and correct e_c errors, where

$$e_d + e_c \le d_{\min} - 1$$

and

$$e_c \le e_d$$

7.5–2 Block and Bit Error Probability for Hard Decision Decoding

In this section we derive bounds on the probability of error for hard decision decoding of linear binary block codes based on error correction only.

From the above discussion, it is clear that the optimum decoder for a binary symmetric channel will decode correctly if (but not necessarily only if) the number of errors in a codeword is less than one-half the minimum distance $d_{\min}$ of the code. That is, any number of errors up to

$$t = \left\lfloor \frac{1}{2}(d_{\min} - 1) \right\rfloor$$

is always correctable. Since the binary symmetric channel is memoryless, the bit errors occur independently. Hence, the probability of m errors in a block of n bits is

$$P(m,n) = \binom{n}{m} p^m (1-p)^{n-m} \tag{7.5–5}$$

and, therefore, the probability of a codeword error is upper-bounded by the expression

$$P_e \le \sum_{m=t+1}^{n} P(m,n) \tag{7.5–6}$$

For high signal-to-noise ratios, i.e., small values of p, Equation 7.5–6 can be approximated by its first term, and we have

$$P_e \approx \binom{n}{t+1} p^{t+1} (1-p)^{n-t-1} \tag{7.5–7}$$

This equation states that when $\mathbf{0}$ is transmitted, the probability of error almost entirely is equal to the probability of receiving sequences of weight $t+1$. To derive an approximate bound on the error probability of each binary symbol in a codeword, we note that if $\mathbf{0}$ is sent and a sequence of weight $t+1$ is received, the decoder will decode the received sequence of weight $t+1$ to a codeword at a distance at most t from the received sequence and hence a distance of at most $2t+1$ from $\mathbf{0}$. But since the minimum weight of the code is $2t+1$, the decoded codeword has to be of weight $2t+1$. This means that for each highly probable block error we have $2t+1$ bit errors in the codeword components; hence from Equation 7.5–7 we obtain

$$P_{bs} \approx \frac{2t+1}{n} \binom{n}{t+1} p^{t+1} (1-p)^{n-t-1} \tag{7.5–8}$$

Equality holds in Equation 7.5–6 if the linear block code is a *perfect code*. To describe the basic characteristics of a perfect code, suppose we place a sphere of radius t around each of the possible transmitted codewords. Each sphere around a codeword contains the set of all codewords of Hamming distance less than or equal to t from the codeword. Now, the number of codewords in a sphere of radius $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ is

$$1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{t} = \sum_{i=0}^{t} \binom{n}{i} \tag{7.5–9}$$

Since there are $M = 2^k$ possible transmitted codewords, there are 2^k nonoverlapping spheres, each having a radius t. The total number of codewords enclosed in the 2^k spheres cannot exceed the 2^n possible received codewords. Thus, a t-error correcting code must satisfy the inequality

$$2^k \sum_{i=0}^{t} \binom{n}{i} \le 2^n \tag{7.5–10}$$

or, equivalently,

$$2^{n-k} \geq \sum_{i=0}^{t} \binom{n}{i} \tag{7.5–11}$$

A perfect code has the property that all spheres of Hamming distance $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$ around the $M = 2^k$ possible transmitted codewords are disjoint and every received codeword falls in one of the spheres. Thus, every received codeword is at most at a distance t from one of the possible transmitted codewords, and Equation 7.5–11 holds with equality. For such a code, all error patterns of weight less than or equal to t are corrected by the optimum (minimum-distance) decoder. On the other hand, any error pattern of weight $t + 1$ or greater cannot be corrected. Consequently, the expression for the error probability given in Equation 7.5–6 holds with equality. The reader can easily verify that the Hamming codes, which have the parameters $n = 2^{n-k} - 1$, $d_{\min} = 3$, and $t = 1$, are an example of perfect codes. The (23, 12) Golay code has parameters $d_{\min} = 7$ and $t = 3$. It can be easily verified that this code is also a perfect code. These two nontrivial codes and the trivial code consisting of two codewords of odd length n and $d_{\min} = n$ are the only perfect binary block codes.

A *quasi-perfect code* is characterized by the property that all spheres of Hamming radius t around the M possible transmitted codewords are disjoint and every received codeword is at most at a distance $t + 1$ from one of the possible transmitted codewords. For such a code, all error patterns of weight less than or equal to t and some error patterns of weight $t + 1$ are correctable, but any error pattern of weight $t + 2$ or greater leads to incorrect decoding of the codeword. Clearly, Equation 7.5–6 is an upper bound on the error probability, and

$$P_e \geq \sum_{m=t+2}^{n} P(m, n) \tag{7.5–12}$$

is a lower bound.

A more precise measure of the performance for quasi-perfect codes can be obtained by making use of the inequality in Equation 7.5–11. That is, the total number of codewords outside the 2^k spheres of radius t is

$$N_{t+1} = 2^n - 2^k \sum_{i=0}^{t} \binom{n}{i}$$

If these codewords are equally subdivided into 2^k sets and each set is associated with one of the 2^k spheres, then each sphere is enlarged by the addition of

$$\beta_{t+1} = 2^{n-k} - \sum_{i=0}^{t} \binom{n}{i} \tag{7.5–13}$$

codewords having distance $t + 1$ from the transmitted codeword. Consequently, of the $\binom{n}{t+1}$ error patterns of distance $t + 1$ from each codeword, we can correct β_{t+1} error patterns. Thus, the error probability for decoding the quasi- perfect code may be expressed as

$$P_e = \sum_{m=t+2}^{n} P(m,n) + \left[\binom{n}{t+1} - \beta_{t+1}\right] p^{t+1}(1-p)^{n-t-1} \tag{7.5–14}$$

Another pair of upper and lower bounds is obtained by considering two codewords that differ by the minimum distance. First, we note that P_e cannot be less than the probability of erroneously decoding the transmitted codeword as its nearest neighbor, which is at a distance $d_{\min}$ from the transmitted codeword. That is,

$$P_e \geq \sum_{m=\lfloor d_{\min}/2 \rfloor + 1}^{d_{\min}} \binom{d_{\min}}{m} p^m (1-p)^{d_{\min}-m} \tag{7.5–15}$$

On the other hand, P_e cannot be greater than $2^k - 1$ times the probability of erroneously decoding the transmitted codeword as its nearest neighbor, which is at a distance $d_{\min}$ from the transmitted codeword. That is a union bound, which is expressed as

$$P_e \leq (2^k - 1) \sum_{m=\lfloor d_{\min}/2 \rfloor + 1}^{d_{\min}} \binom{d_{\min}}{m} p^m (1-p)^{d_{\min}-m} \tag{7.5–16}$$

When $M = 2^k$ is large, the lower bound in Equation 7.5–15 and the upper bound in Equation 7.5–16 are very loose.

General bounds on block and bit error probabilities under hard decision decoding are obtained by using relations derived in Equations 7.2–39, 7.2–43, and 7.2–48. The value of Δ for hard decision decoding was found in Example 6.8–1 and is given by $\Delta = \sqrt{4p(1-p)}$. The results are

$$P_e \leq (A(Z) - 1)\Big|_{Z=\sqrt{4p(1-p)}} \tag{7.5–17}$$

$$P_e \leq (2^k - 1)\left[4p(1-p)\right]^{\frac{d_{\min}}{2}} \tag{7.5–18}$$

$$P_b \leq \frac{1}{k}\frac{\partial}{\partial Y} B(Y,Z)\Big|_{Y=1,Z=\sqrt{4p(1-p)}} \tag{7.5–19}$$

7.6

COMPARISON OF PERFORMANCE BETWEEN HARD DECISION AND SOFT DECISION DECODING

It is both interesting and instructive to compare the bounds on the error rate performance of linear block codes for soft decision decoding and hard decision decoding on an AWGN channel. For illustrative purposes, we use the Golay (23, 12) code, which has the relatively simple weight distribution given in Equation 7.3–15. As stated previously, this code has a minimum distance $d_{\min} = 7$.

First we compute and compare the bounds on the error probability for hard decision decoding. Since the Golay (23, 12) code is a perfect code, the exact error probability for hard decision decoding is given by Equation 7.5–6 as

$$\begin{aligned} P_e &= \sum_{m=4}^{23} \binom{23}{m} p^m (1-p)^{23-m} \\ &= 1 - \sum_{m=0}^{3} \binom{23}{m} p^m (1-p)^{23-m} \end{aligned} \tag{7.6–1}$$

where p is the probability of a binary digit error for the binary symmetric channel. Binary (or four-phase) coherent PSK is assumed to be the modulation/demodulation technique for the transmission and reception of the binary digits contained in each codeword. Thus, the appropriate expression for p is given by Equation 7.5–1. In addition to the exact error probability given by Equation 7.6–1, we have the lower bound given by Equation 7.5–15 and the three upper bounds given by Equations 7.5–16, 7.5–17, and 7.5–18. Numerical results obtained from these bounds are compared with the exact error probability in Figure 7.6–1. We observe that the lower bound is very loose. At $P_e = 10^{-5}$, the lower bound is off by approximately 2 dB from the exact error probability. All three upper bounds are very loose for error rates above $P_e = 10^{-2}$.

It is also interesting to compare the performance between soft and hard decision decoding. For this comparison, we use the upper bounds on the error probability for soft decision decoding given by Equation 7.4–7 and the exact error probability for hard decision decoding given by Equation 7.6–1. Figure 7.6–2 illustrates these performance characteristics. We observe that the two bounds for soft decision decoding differ by approximately 0.5 dB at $P_e = 10^{-6}$ and by approximately 1 dB at $P_e = 10^{-2}$. We also

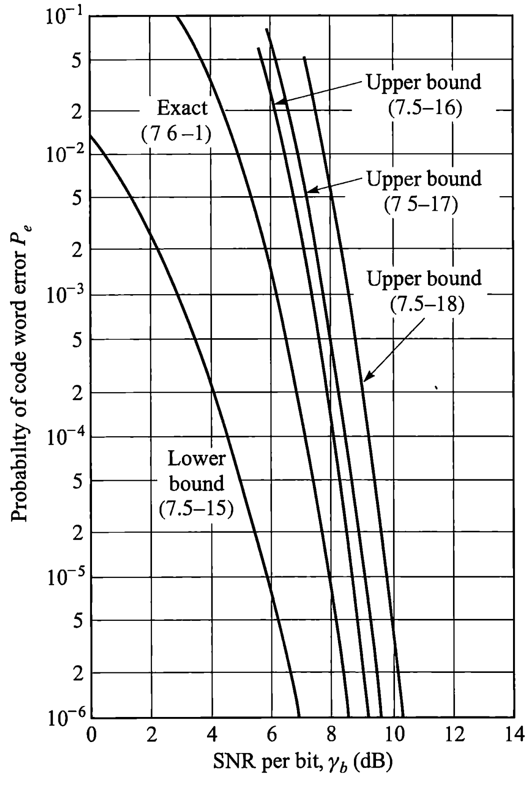


FIGURE 7.6–1
Comparison of bounds with exact error probability for hard decision decoding of Golay (23, 12) code.

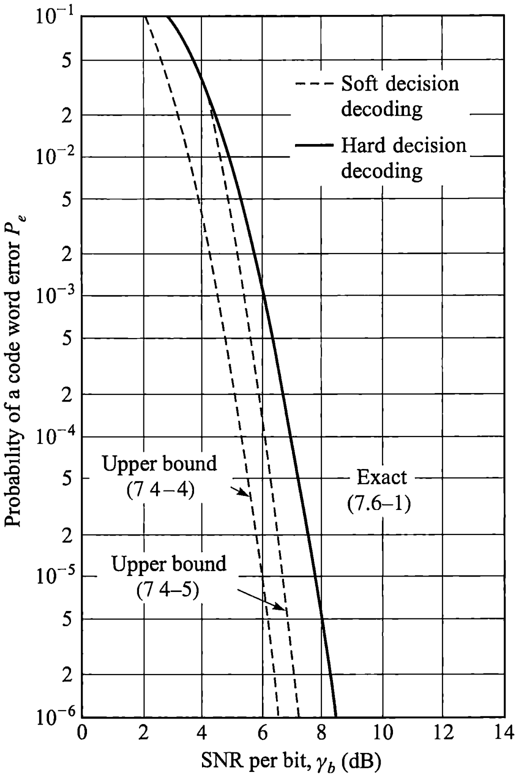


FIGURE 7.6–2
Comparison of soft-decision decoding versus hard-decision decoding for a (23, 12) Golay code.

observe that the difference in performance between hard and soft decision decoding is approximately 2 dB in the range $10^{-2} < P_e < 10^{-6}$. In the range $P_e > 10^{-2}$, the curve of the error probability for hard decision decoding crosses the curves for the bounds. This behavior indicates that the bounds for soft decision decoding are loose when $P_e > 10^{-2}$.

As we observed in Example 6.8–3 and Figure 6.8–4, there exists a roughly 2-dB gap between the cutoff rates of a BPSK modulated scheme under soft and hard decision decoding. A similar gap also exits between the capacities in these two cases. This result can be shown directly by noting that the capacity of a BSC, corresponding to hard decision decoding, is given by Equation 6.5–29 as

$$C = 1 - H_2(p) = 1 + p\log_2 p + (1-p)\log_2(1-p) \tag{7.6–2}$$

where

$$p = Q\left(\sqrt{2\gamma_b R_c}\right) \tag{7.6–3}$$

For small values of R_c we can use the approximation

$$Q(\epsilon) \approx \frac{1}{2} - \frac{\epsilon}{\sqrt{2\pi}} \qquad \epsilon > 0 \tag{7.6–4}$$

to obtain

$$p \approx \frac{1}{2} - \sqrt{\frac{\gamma_b R_c}{\pi}} \tag{7.6–5}$$

Substituting this result into Equation 7.6–2 and using the approximation

$$\log_2(1+x) \approx \frac{x - \frac{1}{2}x^2}{\ln 2} \tag{7.6–6}$$

we obtain

$$C = \frac{2}{\pi \ln 2}\, \gamma_b R_c \tag{7.6–7}$$

Now we set $C = R_c$. Thus, in the limit as R_c approaches zero, we obtain the result

$$\gamma_b = \frac{1}{2}\pi \ln 2 \sim 0.37 \text{ dB} \tag{7.6–8}$$

The capacity of the binary-input AWGN channel with soft decision decoding can be computed in a similar manner. The expression for the capacity in bits per code symbol, derived in Equations 6.5–30 to 6.5–32 can be approximated for low values of R_c as

$$C \approx \frac{\gamma_b R_c}{\ln 2} \tag{7.6–9}$$

Again, we set $C = R_c$. Thus, as $R_c \to 0$, the minimum SNR per bit to achieve capacity is

$$\gamma_b = \ln 2 \sim -1.6 \text{ dB} \tag{7.6–10}$$

Equations 7.6–8 and 7.6–10 clearly show that at low SNR values there exists roughly a 2-dB difference between the performance of hard and soft decision decoding. As seen from Figure 6.8–4, increasing SNR results in a decrease in the performance difference between hard and soft decision decoding. For example, at $R_c = 0.8$, the difference reduces to about 1.5 dB.

The curves in Figure 6.8–4 provide more information than just the difference in performance between soft and hard decision decoding. These curves also specify the minimum SNR per bit that is required for a given code rate. For example, a code rate of $R_c = 0.8$ can provide arbitrarily small error probability at an SNR per bit of 2 dB, when soft decision decoding is used. By comparison, an uncoded binary PSK requires 9.6 dB to achieve an error probability of 10^{-5}. Hence, a 7.6-dB gain is possible by employing a rate $R_c = \frac{4}{5}$ code. This gain is obtained by expanding the bandwidth by 25% since the bandwidth expansion factor of such a code is $1/R_c = 1.25$. To achieve such a large coding gain usually implies the use of an extremely long block length code, and generally a complex decoder. Nevertheless, the curves in Figure 6.8–4 provide a benchmark for comparing the coding gains achieved by practically implementable codes with the ultimate limits for either soft or hard decision decoding.

7.7
BOUNDS ON MINIMUM DISTANCE OF LINEAR BLOCK CODES

The expressions for the probability of error derived in this chapter for soft decision and hard decision decoding of linear binary block codes clearly indicate the importance of the minimum-distance parameter in the performance of the code. If we consider soft decision decoding, for example, the upper bound on the error probability given by Equation 7.4–7 indicates that, for a given code rate $R_c = k/n$, the probability of error in an AWGN channel decreases exponentially with $d_{\min}$. When this bound is used in conjunction with the lower bound on $d_{\min}$ given below, we obtain an upper bound on P_e, the probability of a codeword error. Similarly, we may use the upper bound given by Equation 7.5–6 for the probability of error for hard decision decoding in conjunction with the lower bound on $d_{\min}$ to obtain an upper bound on the error probability for linear binary block codes on the binary symmetric channel.

On the other hand, an upper bound on $d_{\min}$ can be used to determine a lower bound on the probability of error achieved by the best code. For example, suppose that hard decision decoding is employed. In this case, we can use Equation 7.5–15 in conjunction with an upper bound on $d_{\min}$, to obtain a lower bound on P_e for the best (n, k) code. Thus, upper and lower bounds on $d_{\min}$ are important in assessing the capabilities of codes. In this section we study some bounds on minimum distance of linear block codes.

7.7–1 Singleton Bound

The Singleton bound is obtained using the properties of the parity check matrix $\boldsymbol{H}$. Recall from the discussion in Section 7.2–2 that the minimum distance of a linear block code is equal to the minimum number of columns of $\boldsymbol{H}$, the parity check matrix, that are linearly dependent. From this we conclude that the rank of the parity check matrix is equal to $d_{\min} - 1$. Since the parity check matrix is an $(n - k) \times n$ matrix, its rank is at most $n - k$. Hence,

$$d_{\min} - 1 \leq n - k \tag{7.7–1}$$

or

$$d_{\min} \leq n - k + 1 \tag{7.7–2}$$

The bound given in Equation 7.7–2 is called the *Singleton bound*. Since $d_{\min} - 1$ is approximately twice the number of errors that a code can correct, from Equation 7.7–1 we conclude that the number of parity checks in a code must be at least equal to twice the number of errors a code can correct. Although the proof of the Singleton bound presented here was based on the linearity of the code, this bound applies to all block codes, linear and nonlinear, binary and nonbinary.

Codes for which the Singleton bound is satisfied with equality, i.e., codes for which $d_{\min} = n - k + 1$, are called *maximum-distance separable*, or MDS, codes. Repetition codes and their duals are examples of MDS codes. In fact these codes are the only binary MDS codes.[†] In the class of nonbinary codes, Reed-Solomon codes studied in Section 7.11 are the most important examples of MDS codes.

Dividing both sides of the Singleton bound by n, we have

$$\frac{d_{\min}}{n} \leq 1 - R_c + \frac{1}{n} \tag{7.7–3}$$

If we define

$$\delta_n = \frac{d_{\min}}{n} \tag{7.7–4}$$

we have

$$\delta_n \leq 1 - R_c + \frac{1}{n} \tag{7.7–5}$$

Note that $d_{\min}/2$ is roughly the number of errors that a code can correct. Therefore,

$$\frac{1}{2}\delta_n \approx \frac{t}{n} \tag{7.7–6}$$

i.e., $\frac{\delta_n}{2}$ approximately represents the fraction of correctable errors in transmission of n bits.

If we define $\delta = \lim_{n\to\infty} \delta_n$, we conclude that as $n \to \infty$,

$$\delta \leq 1 - R_c \tag{7.7–7}$$

This is the asymptotic form of the Singleton bound.

7.7–2 Hamming Bound

The *Hamming* or *sphere packing* bound was previously developed in our study of the performance of hard decision decoding and is given by Equation 7.5–11 as

$$2^{n-k} \geq \sum_{i=0}^{t} \binom{n}{i} \tag{7.7–8}$$

Taking the logarithm and dividing by n result in

$$1 - R_c \geq \frac{1}{n} \log_2 \left[\sum_{i=0}^{t} \binom{n}{i} \right] \tag{7.7–9}$$

or

$$1 - R_c \geq \frac{1}{n} \log_2 \left[\sum_{i=0}^{\left\lfloor \frac{d_{\min}-1}{2} \right\rfloor} \binom{n}{i} \right] \tag{7.7–10}$$

This relation gives an upper bound for $d_{\min}$ in terms of n and k, known as the Hamming bound. Note that the proof of the Hamming bound is independent of the linearity of the code; therefore this bound applies to all block codes. For the q-ary block codes the Hamming bound yields

---

[†] The (n, n) code with $d_{\min} = 1$ is another MDS code, but this code introduces no redundancy and can hardly be called a code.

$$1 - R_c \geq \frac{1}{n} \log_q \left[\sum_{i=0}^{t} \binom{n}{i} (q-1)^i \right] \tag{7.7–11}$$

In Problem 7.39 it is shown that for large n the right-hand side of Equation 7.7–9 can be approximated by

$$\sum_{i=0}^{t} \binom{n}{i} \approx 2^{nH_b(\frac{t}{n})} \tag{7.7–12}$$

where $H_b(\cdot)$ is the binary entropy function defined in Equation 6.2–6. Using this approximation, and Equation 7.7–6, we see that the asymptotic form of the Hamming bound for binary codes becomes

$$H_b\left(\frac{\delta}{2}\right) \leq 1 - R_c \tag{7.7–13}$$

The Hamming bound is tight for high-rate codes.

As discussed before, a code satisfying the Hamming bound given by Equation 7.7–10 with equality is called a perfect code. It has been shown by Tietäväinen (1973) that the only binary perfect codes† are repetition codes with odd length, Hamming codes, and the (23, 12) Golay code with minimum distance 7. There exists only one nonbinary perfect code which is the (11,6) ternary Golay code with minimum distance 5.

7.7–3 Plotkin Bound

The *Plotkin bound* due to Plotkin (1960) states that for any q-ary block code we have

$$\frac{d_{\min}}{n} \leq \frac{q^k - q^{k-1}}{q^k - 1} \tag{7.7–14}$$

For binary codes this bound becomes

$$d_{\min} \leq \frac{n2^{k-1}}{2^k - 1} \tag{7.7–15}$$

The proof of the Plotkin bound for binary linear block codes is given in Problem 7.40. The proof is based on noting that the minimum distance of a code cannot exceed its average codeword weight.

The form of the Plotkin bound given in Equation 7.7–15 is effective for low rates. Another version of the Plotkin bound, given in Equation 7.7–16 for binary codes, is tighter for higher-rate codes:

$$d_{\min} \leq \min_{1 \leq j \leq k} (n - k + j) \frac{2^{j-1}}{2^j - 1} \tag{7.7–16}$$

---

†Here again an $(n, 1)$ code can be considered as a trivial perfect code.

A simplified version of this bound, obtained by choosing $j = 1 + \lfloor \log_2 d_{\min} \rfloor$, results in

$$2d_{\min} - 2 - \lfloor \log_2 d_{\min} \rfloor \leq n - k \tag{7.7–17}$$

The asymptotic form of this bound with the assumption of $\delta \leq \frac{1}{2}$ is

$$\delta \leq \frac{1}{2}(1 - R_c) \tag{7.7–18}$$

7.7–4 Elias Bound

The asymptotic form of the *Elias bound* (see Berlekamp (1968)) states that for any binary code with $\delta \leq \frac{1}{2}$ we have

$$H_b\left(\frac{1}{2}\left(1 - \sqrt{1 - 2\delta}\right)\right) \leq 1 - R_c \tag{7.7–19}$$

The Elias bound also applies to nonbinary codes. For nonbinary codes this bound states that for any q-ary code with $\delta \leq 1 - \frac{1}{q}$ we have

$$H_q\left(\frac{q-1}{q}\left(1 - \sqrt{1 - \frac{q}{q-1}\delta}\right)\right) \leq 1 - R_c \tag{7.7–20}$$

where $H_q(\cdot)$ is defined by

$$H_q(p) = -p \log_q p - (1 - p) \log_q (1 - p) + p \log_q (q - 1) \tag{7.7–21}$$

for $0 \leq p \leq 1$.

7.7–5 McEliece-Rodemich-Rumsey-Welch (MRRW) Bound

The *McEliece-Rodemich-Rumsey-Welch (MRRW) bound* derived by McEliece et al. (1977) is the tightest known bound for low to moderate rates. This bound has two forms; the simpler form has the asymptotic form given by

$$R_c \leq H_b\left(\frac{1}{2} - \sqrt{\delta(1 - \delta)}\right) \tag{7.7–22}$$

for binary codes and for $\delta \leq \frac{1}{2}$. This bound is derived based on linear programming techniques.

7.7–6 Varshamov-Gilbert Bound

All bounds stated so far give the *necessary* conditions that must be stratified by the three main parameters n, k, and d of a block code. The *Varshamov-Gilbert bound* due to Gilbert (1952) and Varshamov (1957) gives the *sufficient conditions* for the existence of an (n, k) code with minimum distance $d_{\min}$. The Varshamov-Gilbert bound in fact goes further to prove the existence of a *linear* block code with the given parameters.

The Varshamov-Gilbert states that if the inequality

$$\sum_{i=0}^{d-2} \binom{n-1}{i} (q-1)^i < q^{n-k} \qquad (7.7\text{–}23)$$

is satisfied, then there exists a q-ary (n, k) linear block code with minimum distance $d_{\min} \geq d$. For the binary case the Varshamov-Gilbert bound becomes

$$\sum_{i=0}^{d-2} \binom{n-1}{i} < 2^{n-k} \qquad (7.7\text{–}24)$$

The asymptotic version of the Varshamov-Gilbert bound states that if for $0 < \delta \leq 1 - \frac{1}{q}$ we have

$$H_q(\delta) < 1 - R_c \qquad (7.7\text{–}25)$$

where $H_q(\cdot)$ is given by Equation 7.7–21, then there exists a q-ary $(n,\ R_c n)$ linear block code with minimum distance of at least δn.

A comparison of the asymptotic version of the bounds discussed above is shown in Figure 7.7–1 for the binary codes. As seen in the figure, the tightest asymptotic upper bounds are the Elias and the MRRW bounds. We add here that there exists a second

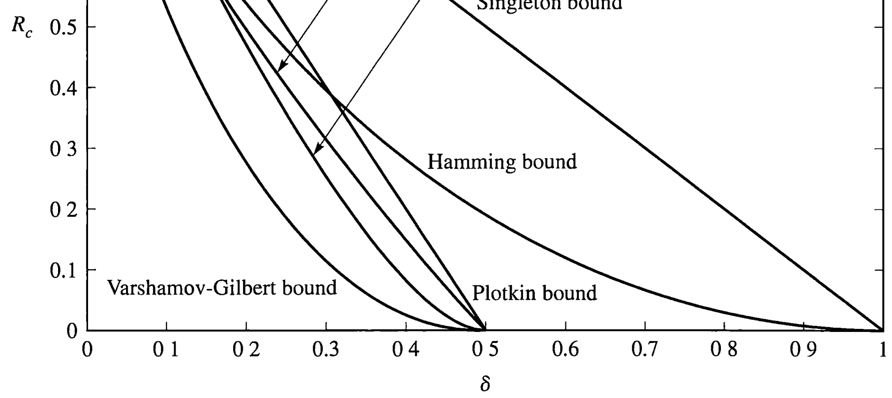


FIGURE 7.7–1
Comparison of Asymptotic Bounds.

version of the MRRW bound that is better than the Elias bound at higher rates. The ordering of the bounds shown on this plot is only an indication of how these bounds compare as $n \to \infty$. The region between the tightest upper bound and the Varshamov-Gilbert lower bound can still be a rather wide region for certain block lengths. For instance, for a (127, 33) code the best upper bound and lower bound yield $d_{\min} = 48$ and $d_{\min} = 32$, respectively (Verhoeff (1987)).

7.8
MODIFIED LINEAR BLOCK CODES

In many cases design techniques for linear block codes result in codes with certain parameters that might not be the exact parameters that are required for a certain application. For example, we have seen that for Hamming codes $n = 2^m - 1$ and $d_{\min} = 3$. In Section 7.10, we will see that the codeword lengths of BCH codes, which are widely used block codes, are equal to $2^m - 1$. Therefore, in many cases in order to change the parameters of a code, the code has to be modified. In this section we study main methods for modification of linear block codes.

7.8–1 Shortening and Lengthening

Let us assume $\mathcal{C}$ is an (n, k) linear block code with minimum distance $d_{\min}$. *Shortening* of $\mathcal{C}$ means choosing some $1 \le j < k$ and considering only 2^{k-j} information sequences whose leading j bits are zero. Since these components carry no information, they can be deleted. The result is a *shortened* code. The resulting code is a systematic $(n - j, k - j)$ linear block code with rate $R_c = \frac{k-j}{n-j}$ which is less than the rate of the original code. Since the codewords of a shortened code are the result of removing j zeros for the codewords of $\mathcal{C}$, the minimum weight of the shortened code is at least as large as the minimum weight of the original code. If j is large, the minimum weight of the shortened code is usually larger than the minimum weight of the original code.

EXAMPLE 7.8–1. A (15, 11) Hamming code can be shortened by 3 bits to obtain a (12, 8) shortened Hamming code which is 8 bits (1 byte) of information. The (15, 11) can also be shortened by 7 bits to obtain an (8, 4) shortened Hamming code with parity check matrix

$$\boldsymbol{H} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{7.8–1}$$

This code has a minimum distance of 4.

EXAMPLE 7.8–2. Consider an (8, 4) linear block code with generator and parity check matrices given by

$$\begin{aligned}\boldsymbol{G}&=\begin{bmatrix}1&1&1&1&1&1&1&1\\0&1&0&1&1&1&0&0\\0&0&1&0&1&1&1&0\\0&0&0&1&0&1&1&1\end{bmatrix}\\ \boldsymbol{H}&=\begin{bmatrix}1&1&1&1&1&1&1&1\\0&0&0&1&0&1&1&1\\0&0&1&0&1&1&1&0\\0&1&0&0&1&0&1&1\end{bmatrix}\end{aligned}\tag{7.8–2}$$

Shortening this code by 1 bit results in a (7, 3) linear block code with the following generator and parity check matrices.

$$\begin{aligned}\boldsymbol{G}&=\begin{bmatrix}1&0&1&1&1&0&0\\0&1&0&1&1&1&0\\0&0&1&0&1&1&1\end{bmatrix}\\ \boldsymbol{H}&=\begin{bmatrix}1&1&1&1&1&1&1\\0&0&1&0&1&1&1\\0&1&0&1&1&1&0\\1&0&0&1&0&1&1\end{bmatrix}\end{aligned}\tag{7.8–3}$$

Both codes have a minimum distance of 4.

Shortened codes are used in a variety of applications. One example is the shortened Reed-Solomon codes used in CD recording where a (255, 251) Reed-Solomon code is shortened to a (32, 28) code.

Lengthening a code is the inverse of the shortening operation. Here j extra information bits are added to the code to obtain an $(n+j, k+j)$ linear block code. The rate of the lengthened code is higher than that of the original code, and its minimum distance cannot exceed the minimum distance of the original code. Obviously in the process of shortening and lengthening, the number of parity check bits of a code does not change. In Example 7.8–2 the (8, 4) code can be considered a lengthened version of the (7, 3) code.

7.8–2 Puncturing and Extending

Puncturing is a popular technique to increase the rate of a low-rate code. In puncturing an (n, k) code the number of information bits k remains unchanged whereas some components of the code are deleted (punctured). The result is an $(n-j, k)$ linear block code with higher rate and possibly lower minimum distance. Obviously the minimum distance of a punctured code cannot be higher than the minimum distance of the original code.

EXAMPLE 7.8–3. The (8, 4) code of Example 7.8–2 can be punctured to obtain a (7, 4) code with

$$\begin{aligned}
G &= \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \\
H &= \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}
\end{aligned} \qquad (7.8\text{–}4)$$

The reverse of puncturing is *extending* a code. In extending a code, while k remains fixed, more parity check bits are added. The rate of the resulting code is lower, and the resulting minimum distance is at least as large as that of the original code.

EXAMPLE 7.8–4. A (7, 4) Hamming code can be extended by adding an overall parity check bit. The resulting code is an (8, 4) extended Hamming code whose parity check matrix has a row of all 1s to check the overall parity. If the parity check matrix of the original Hamming code is an $(n-k) \times n$ matrix $\boldsymbol{H}$, the parity check matrix of the extended Hamming code is given by

$$\boldsymbol{H}_{\mathrm{e}} = \begin{bmatrix} \boldsymbol{H} & \vdots & \boldsymbol{0} \\ \cdots & \cdots & \cdots \\ \boldsymbol{1} & \vdots & 1 \end{bmatrix} \qquad (7.8\text{–}5)$$

where $\boldsymbol{1}$ denotes a $1 \times n$ row vector of 1s and $\boldsymbol{0}$ denotes a $(n-k) \times 1$ vector column of 0s.

7.8–3 Expurgation and Augmentation

In these two modifications of a code, the block length n remains unchanged, and the number of information sequence k is decreased in *expurgation* and increased in *augmentation*.

The result of expurgation of an (n, k) linear block code is an $(n, k-j)$ code with lower rate whose minimum distance is guaranteed to be at least equal to the minimum distance of the original code. This can be done by eliminating j rows of the generator matrix $\boldsymbol{G}$. The process of augmentation is the reverse of expurgation in which $2^j (n, k)$ codes are merged to generate an $(n, k+j)$ code.

7.9

CYCLIC CODES

Cyclic codes are an important class of linear block codes. Additional structure built in the cyclic code family makes their algebraic decoding at reduced computational complexity possible. The important class of BCH codes and Reed-Solomon (RS) codes belongs to the class of cyclic codes. Cyclic codes were first introduced by Prange (1957).

7.9–1 Cyclic Codes — Definition and Basic Properties

Cyclic codes are a subset of the class of linear block codes that satisfy the following cyclic shift property: if $\boldsymbol{c} = (c_{n-1}\, c_{n-2} \cdots c_1\, c_0)$ is a codeword of a cyclic code, then $(c_{n-2}\, c_{n-3} \cdots c_0\, c_{n-1})$, obtained by a cyclic shift of the elements of $\boldsymbol{c}$, is also a codeword. That is, all cyclic shifts of $\boldsymbol{c}$ are codewords. As a consequence of the cyclic property, the codes possess a considerable amount of structure which can be exploited in the encoding and decoding operations. A number of efficient encoding and hard decision decoding algorithms have been devised for cyclic codes that make it possible to implement long block codes with a large number of codewords in practical communication systems. Our primary objective is to briefly describe a number of characteristics of cyclic codes, with emphasis on two important classes of cyclic codes, the BCH and Reed-Solomon codes.

In dealing with cyclic codes, it is convenient to associate with a codeword $\boldsymbol{c} = (c_{n-1}\, c_{n-2} \cdots c_1\, c_0)$ a polynomial $c(X)$ of degree at most $n-1$, defined as

$$c(X) = c_{n-1}X^{n-1} + c_{n-2}X^{n-2} + \cdots + c_1 X + c_0 \tag{7.9–1}$$

For a binary code, each of the coefficients of the polynomial is either 0 or 1.

Now suppose we form the polynomial

$$Xc(X) = c_{n-1}X^{n} + c_{n-2}X^{n-1} + \cdots + c_1 X^2 + c_0 X$$

This polynomial cannot represent a codeword, since its degree may be equal to n (when $c_{n-1} = 1$). However, if we divide $Xc(X)$ by $X^n + 1$, we obtain

$$\frac{Xc(X)}{X^n+1} = c_{n-1} + \frac{c^{(1)}(X)}{X^n+1} \tag{7.9–2}$$

where

$$c^{(1)}(X) = c_{n-2}X^{n-1} + c_{n-3}X^{n-2} + \cdots + c_0 X + c_{n-1}$$

Note that the polynomial $c^{(1)}(X)$ represents the codeword $\boldsymbol{c}^{(1)} = (c_{n-2} \cdots c_0\; c_{n-1})$, which is just the codeword $\boldsymbol{c}$ shifted cyclicly by one position. Since $c^{(1)}(X)$ is the remainder obtained by dividing $Xc(X)$ by $X^n + 1$, we say that

$$c^{(1)}(X) = Xc(X) \quad \bmod (X^n + 1) \tag{7.9–3}$$

In a similar manner, if $c(X)$ represents a codeword in a cyclic code, then $X^i c(X)$ mod $(X^n + 1)$ is also a codeword of the cyclic code. Thus we may write

$$X^i c(X) = Q(X)(X^n + 1) + c^{(i)}(X) \tag{7.9–4}$$

where the remainder polynomial $c^{(i)}(X)$ represents a codeword of the cyclic code, corresponding to i cyclic shifts of $\boldsymbol{c}$ to the right, and $Q(X)$ is the quotient.

We can generate a cyclic code by using a *generator polynomial* $g(X)$ of degree $n-k$. The generator polynomial of an (n, k) cyclic code is a factor of $X^n + 1$ and has the general form

$$g(X) = X^{n-k} + g_{n-k-1}X^{n-k-1} + \cdots + g_1 X + 1 \tag{7.9–5}$$

We also define a *message polynomial* $u(X)$

$$u(X) = u_{k-1}X^{k-1} + u_{k-2}X^{k-2} + \cdots + u_1X + u_0 \tag{7.9–6}$$

where $(u_{k-1}\, u_{k-2} \cdots u_1, u_0)$ represent the k information bits. Clearly, the product $u(X)g(X)$ is a polynomial of degree less than or equal to $n-1$, which may represent a codeword. We note that there are 2^k polynomials $\{u_i(X)\}$, and hence there are 2^k possible codewords that can be formed from a given $g(X)$.

Suppose we denote these codewords as

$$c_m(X) = u_m(X)g(X), \qquad m = 1, 2, \ldots, 2^k \tag{7.9–7}$$

To show that the codewords in Equation 7.9–7 satisfy the cyclic property, consider any codeword $c(X)$ in Equation 7.9–7. A cyclic shift of $c(X)$ produces

$$c^{(1)}(X) = Xc(X) + c_{n-1}(X^n + 1) \tag{7.9–8}$$

and since $g(X)$ divides both $X^n + 1$ and $c(X)$, it also divides $c^{(1)}(X)$; i.e., $c^{(1)}(X)$ can be represented as

$$c^{(1)}(X) = u_1(X)g(X)$$

Therefore, a cyclic shift of any codeword $c(X)$ generated by Equation 7.9–7 yields another codeword.

From the above, we see that codewords possessing the cyclic property can be generated by multiplying the 2^k message polynomials with a unique polynomial $g(X)$, called the generator polynomial of the (n, k) cyclic code, which divides $X^n + 1$ and has degree $n - k$. The cyclic code generated in this manner is a subspace S_c of the vector space S. The dimension of S_c is k.

It is clear from above that an (n, k) cyclic code can exist only if we can find a polynomial $g(X)$ of degree $n - k$ that divides $X^n + 1$. Therefore the problem of designing cyclic codes is equivalent to the problem of finding factors of $X^n + 1$. We have studied this problem for the case where $n = 2^m - 1$ for some positive integer m in the discussion following Equation 7.1–18, and we have seen that for this case the factors of $X^n + 1$ are the minimal polynomials corresponding to the conjugacy classes of nonzero elements of GF(2^m). For general n, the study of the factorization of $X^n + 1$ is more involved. The interested reader is referred to the book by Wicker (1995). Table 7.9–1 presents factoring of $X^n + 1$. The representation in this table is in octal form; therefore the polynomial $X^3 + X^2 + 1$ is represented as 001101 which is equivalent to 15 in octal notation.

EXAMPLE 7.9–1. Consider a code with block length $n = 7$. The polynomial $X^7 + 1$ has the following factors:

$$X^7 + 1 = (X + 1)(X^3 + X^2 + 1)(X^3 + X + 1) \tag{7.9–9}$$

To generate a $(7, 4)$ cyclic code, we may take as a generator polynomial one of the following two polynomials:

$$g_1(X) = X^3 + X^2 + 1$$

TABLE 7.9–1
Factors of $X^n + 1$ Based on MacWilliams and Sloane (1977)

n	Factors
7	3.15.13
9	3.7.111
15	3.7.31.23.37
17	3.471.727
21	3.7.15.13.165.127
23	3.6165.5343
25	3.37.4102041
27	3.7.111.1001001
31	3.51.45.75.73.67.57
33	3.7.2251.3043.3777
35	3.15.13.37.16475.13627
39	3.7.17075.13617.17777
41	3.5747175.6647133
43	3.47771.52225.64213
45	3.7.31.23.27.111.11001.10011
47	3.75667061.43073357
49	3.15.13.10040001.10000201
51	3.7.661.471.763.433.727.637
55	3.37.3777.7164555.5551347
57	3.7.1341035.1735357.1777777
63	3.7.15.13.141.111.165.155.103.163.133.147.127
127	3.301.221.361.211.271.345.325.235.375.203.323.313.253.247.367.217.357.277

and

$$g_2(X) = X^3 + X + 1$$

The codes generated by $g_1(X)$ and $g_2(X)$ are equivalent. The codewords in the $(7, 4)$ code generated by $g_1(X) = X^3 + X^2 + 1$ are given in Table 7.9–2.

EXAMPLE 7.9–2. To determine the possible values of k for a cyclic code with block length $n = 25$, we use Table 7.9–1. From this table, factors of $X^{25} + 1$ are 3, 37, and 4102041 which correspond to $X+1$, $X^4+X^3+X^2+X+1$, and $X^{20}+X^{15}+X^{10}+X^5+1$. The possible (nontrivial) values for $n - k$ are 1, 4, 20, and 5, 21, 24, where the latter three are obtained by multiplying pairs of the polynomials. These correspond to the values 24, 21, 20, 5, 4, and 1 for k.

In general, the polynomial $X^n + 1$ may be factored as

$$X^n + 1 = g(X)h(X)$$

where $g(X)$ denotes the generator polynomial for the (n, k) cyclic code and $h(X)$ denotes the *parity check polynomial* that has degree k. The latter may be used to generate the dual code. For this purpose, we define the *reciprocal polynomial* of $h(X)$ as

$$\begin{aligned} X^k h(X^{-1}) &= X^k(X^{-k} + h_{k-1}X^{-k+1} + h_{k-2}X^{-k+2} + \cdots + h_1 X^{-1} + 1) \\ &= 1 + h_{k-1}X + h_{k-2}X^2 + \cdots + h_1 X^{k-1} + X^k \end{aligned} \tag{7.9–10}$$

TABLE 7.9–2
The (7, 4) Cyclic Code with Generator Polynomial $g_1(X) = X^3 + X^2 + 1$

Information Bits				Codewords						
X^3	X^2	X^1	X^0	X^6	X^5	X^4	X^3	X^2	X^1	X^0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	0	0	1	0	1	1	1
0	1	0	0	0	1	1	0	1	0	0
0	1	0	1	0	1	1	1	0	0	1
0	1	1	0	0	1	0	1	1	1	0
0	1	1	1	0	1	0	0	0	1	1
1	0	0	0	1	1	0	1	0	0	0
1	0	0	1	1	1	0	0	1	0	1
1	0	1	0	1	1	1	0	0	1	0
1	0	1	1	1	1	1	1	1	1	1
1	1	0	0	1	0	1	1	1	0	0
1	1	0	1	1	0	1	0	0	0	1
1	1	1	0	1	0	0	0	1	1	0
1	1	1	1	1	0	0	1	0	1	1

Clearly, the reciprocal polynomial is also a factor of $X^n + 1$. Hence, $X^k h(X^{-1})$ is the generator polynomial of an $(n, n-k)$ cyclic code. This cyclic code is the dual code to the (n, k) code generated from $g(X)$. Thus, the $(n, n-k)$ dual code constitutes the null space of the (n, k) cyclic code.

EXAMPLE 7.9–3. Let us consider the dual code to the (7, 4) cyclic code generated in Example 7.9–1. This dual code is a (7, 3) cyclic code associated with the parity polynomial

$$\begin{aligned} h_1(X) &= (X+1)(X^3+X+1) \\ &= X^4 + X^3 + X^2 + 1 \end{aligned} \tag{7.9–11}$$

The reciprocal polynomial is

$$X^4 h_1(X^{-1}) = 1 + X + X^2 + X^4$$

This polynomial generates the (7, 3) dual code given in Table 7.9–3. The reader can verify that the codewords in the (7, 3) dual code are orthogonal to the codewords in the (7, 4) cyclic code of Example 7.9–1. Note that neither the (7, 4) nor the (7, 3) codes are systematic.

It is desirable to show how a generator matrix can be obtained from the generator polynomial of a cyclic (n, k) code. As previously indicated, the generator matrix for an (n, k) code can be constructed from any set of k linearly independent codewords. Hence, given the generator polynomial $g(X)$, an easily generated set of k linearly independent codewords is the codewords corresponding to the set of k linearly

TABLE 7.9–3
The (7, 3) Dual Code with Generator Polynomial $X^4 h_1(X^{-1}) = X^4 + X^2 + X + 1$

Information Bits			Codewords						
X^2	X^1	X^0	X^6	X^5	X^4	X^3	X^2	X^1	X^0
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	1	1	1
0	1	0	0	1	0	1	1	1	0
0	1	1	0	1	1	1	0	0	1
1	0	0	1	0	0	1	1	0	0
1	0	1	1	0	1	1	0	1	1
1	1	0	1	1	0	0	0	1	0
1	1	1	1	1	1	0	1	0	1

independent polynomials

$$X^{k-1}g(X),\ X^{k-2}g(X),\ Xg(X),\ g(X)$$

Since any polynomial of degree less than or equal to $n-1$ and divisible by $g(X)$ can be expressed as a linear combination of this set of polynomials, the set forms a basis of dimension k. Consequently, the codewords associated with these polynomials form a basis of dimension k for the (n, k) cyclic code.

EXAMPLE 7.9–4. The four rows of the generator matrix for the (7, 4) cyclic code with generator polynomial $g_1(X) = X^3 + X^2 + 1$ are obtained from the polynomials

$$X^i g_1(X) = X^{3+i} + X^{2+i} + X^i, \qquad i = 3, 2, 1, 0$$

It is easy to see that the generator matrix is

$$\boldsymbol{G}_1 = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{7.9–12}$$

Similarly, the generator matrix for the (7, 4) cyclic code generated by the polynomial $g_2(X) = X^3 + X + 1$ is

$$\boldsymbol{G}_2 = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{7.9–13}$$

The parity check matrices corresponding to $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ can be constructed in the same manner by using the respective reciprocal polynomials (see Problem 7.46).

Shortened Cyclic Codes

From Example 7.9–2 and Table 7.9–1 it is clear that we cannot design cyclic (n, k) codes for all values of n and k. One common approach to designing cyclic codes with given parameters is to begin with the design of an (n, k) cyclic code and then shorten it by j bits to obtain an $(n-j, k-j)$ code. The shortening of the cyclic code is carried out by equating the j leading bits of the information sequence to zero and not transmitting them. The resulting codes are called *shortened cyclic codes*, although in general they are not cyclic codes. Of course by adding the deleted j zero bits at the receiver, we can decode these codes with any decoder designed for the original cyclic code.

Shortened cyclic codes are extensively used in the form of shortened Reed-Solomon codes and *cyclic redundancy check* (CRC) codes, which are widely used for error detection in computer communication networks. For more details on CRC codes, see Castagnoli et al. (1990) and Castagnoli et al. (1993).

7.9–2 Systematic Cyclic Codes

Note that the generator matrix obtained by this construction is not in systematic form. We can construct the generator matrix of a cyclic code in the systematic form

$$\boldsymbol{G} = \left[\boldsymbol{I}_k \vdots \boldsymbol{P}\right]$$

from the generator polynomial as follows. First, we observe that the lth row of $\boldsymbol{G}$ corresponds to a polynomial of the form $X^{n-l} + R_l(X)$, $l = 1, 2, \ldots, k$, where $R_l(X)$ is a polynomial of degree less than $n-k$. This form can be obtained by dividing X^{n-l} by $g(X)$. Thus, we have

$$\frac{X^{n-l}}{g(X)} = Q_l(X) + \frac{R_l(X)}{g(X)}, \qquad l = 1, 2, \ldots, k$$

or, equivalently,

$$X^{n-l} = Q_l(X)g(X) + R_l(X), \qquad l = 1, 2, \ldots, k \tag{7.9–14}$$

where $Q_l(X)$ is the quotient. But $X^{n-l} + R_l(X)$ is a codeword of the cyclic code since $X^{n-l} + R_l(X) = Q_l(X)g(X)$. Therefore the desired polynomial corresponding to the lth row of $\boldsymbol{G}$ is $X^{n-l} + R_l(X)$.

EXAMPLE 7.9–5. For the (7,4) cyclic code with generator polynomial $g_2(X) = X^3 + X + 1$, previously discussed in Example 7.9–4, we have

$$\begin{aligned}
X^6 &= (X^3 + X + 1)g_2(X) + X^2 + 1 \\
X^5 &= (X^2 + 1)g_2(X) + X^2 + X + 1 \\
X^4 &= Xg_2(X) + X^2 + X \\
X^3 &= g_2(X) + X + 1
\end{aligned}$$

Hence, the generator matrix of the code in systematic form is

$$\boldsymbol{G}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix} \tag{7.9–15}$$

and the corresponding parity check matrix is

$$\boldsymbol{H}_2 = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{7.9–16}$$

It is left as an exercise for the reader to demonstrate that the generator matrix $\boldsymbol{G}_2$ given by Equation 7.9–13 and the systematic form given by Equation 7.9–15 generate the same set of codewords (see Problem 7.16).

The method for constructing the generator matrix $\boldsymbol{G}$ in systematic form according to Equation 7.9–14 also implies that a systematic code can be generated directly from the generator polynomial $g(X)$. Suppose that we multiply the message polynomial $u(X)$ by X^{n-k}. Thus, we obtain

$$X^{n-k}u(X) = u_{k-1}X^{n-1} + u_{k-2}X^{n-2} + \cdots + u_1X^{n-k+1} + u_0X^{n-k}$$

In a systematic code, this polynomial represents the first k bits in the codeword $c(X)$. To this polynomial we must add a polynomial of degree less than $n-k$ representing the parity check bits. Now, if $X^{n-k}u(X)$ is divided by $g(X)$, the result is

$$\frac{X^{n-k}u(X)}{g(X)} = Q(X) + \frac{r(X)}{g(X)}$$

or, equivalently,

$$X^{n-k}u(X) = Q(X)g(X) + r(X) \tag{7.9–17}$$

where $r(X)$ has degree less than $n-k$. Clearly, $Q(X)g(X)$ is a codeword of the cyclic code. Hence, by adding (modulo-2) $r(X)$ to both sides of Equation 7.9–17, we obtain the desired systematic code.

To summarize, the systematic code may be generated by

1. Multiplying the message polynomial $u(X)$ by X^{n-k}
2. Dividing $X^{n-k}u(X)$ by $g(X)$ to obtain the remainder $r(X)$
3. Adding $r(X)$ to $X^{n-k}u(X)$

Below we demonstrate how these computations can be performed by using shift registers with feedback.

Since $X^n + 1 = g(X)h(X)$ or, equivalently, $g(X)h(X) = 0 \bmod (X^n + 1)$, we say that the polynomials $g(X)$ and $h(X)$ are *orthogonal*. Furthermore, the polynomials $X^i g(X)$ and $X^j h(X)$ are also orthogonal for all i and j. However, the vectors corresponding to the polynomials $g(X)$ and $h(X)$ are orthogonal only if the ordered elements of one of these vectors are reversed. The same statement applies to the vectors corresponding to $X^i g(X)$ and $X^j h(X)$. In fact, if the parity polynomial $h(X)$ is used as a generator for the $(n, n-k)$ dual code, the set of codewords obtained just comprises the same codewords generated by the reciprocal polynomial except that the code vectors are reversed. This implies that the generator matrix for the dual code obtained from the reciprocal polynomial $X^k h(X^{-1})$ can also be obtained indirectly from $h(X)$. Since the parity check matrix $\boldsymbol{H}$ for the (n, k) cyclic code is the generator matrix for the dual code, it follows that $\boldsymbol{H}$ can also be obtained from $h(X)$. The following example illustrates these relationships.

EXAMPLE 7.9–6. The dual code to the (7, 4) cyclic code generated by $g_1(X) = X^3 + X^2 + 1$ is the (7, 3) dual code that is generated by the reciprocal polynomial $X^4 h_1(X^{-1}) = X^4 + X^2 + X + 1$. However, we may also use $h_1(X)$ to obtain the generator matrix for the dual code. Then the matrix corresponding to the polynomials $X^i h_1(X)$, $i = 2, 1, 0$, is

$$\boldsymbol{G}_{h1} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}$$

The generator matrix for the (7, 3) dual code, which is the parity check matrix for the (7, 4) cyclic code, consists of the rows of $\boldsymbol{G}_{h1}$ taken in reverse order. Thus,

$$\boldsymbol{H}_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

The reader may verify that $\boldsymbol{G}_1 \boldsymbol{H}_1^t = \boldsymbol{0}$. Note that the column vectors of $\boldsymbol{H}_1$ consist of all seven binary vectors of length 3, except the all-zero vector. But this is just the description of the parity check matrix for a (7, 4) Hamming code. Therefore, the (7, 4) cyclic code is equivalent to the (7, 4) Hamming code.

7.9–3 Encoders for Cyclic Codes

The encoding operations for generating a cyclic code may be performed by a linear feedback shift register based on the use of either the generator polynomial or the parity polynomial. First, let us consider the use of $g(X)$.

As indicated above, the generation of a systematic cyclic code involves three steps, namely, multiplying the message polynomial $u(X)$ by X^{n-k}, dividing the product by $g(X)$, and adding the remainder to $X^{n-k}u(X)$. Of these three steps, only the division is nontrivial.

The division of the polynomial $A(X) = X^{n-k}u(X)$ of degree $n-1$ by the polynomial

$$g(X) = g_{n-k}X^{n-k} + g_{n-k-1}X^{n-k-1} + \cdots + g_1 X + g_0$$

may be accomplished by the $(n-k)$-stage feedback shift register illustrated in Figure 7.9–1. Initially, the shift register contains all zeros. The coefficients of $A(X)$ are clocked into the shift register one (bit) coefficient at a time, beginning with the higher-order coefficients, i.e., with a_{n-1}, followed by a_{n-2}, and so on. After the kth shift, the first nonzero output of the quotient is $q_{k-1} = g_{n-k}a_{n-1}$. Subsequent outputs are generated as illustrated in Figure 7.9–1. For each output coefficient in the quotient, we must subtract the polynomial $g(X)$ multiplied by that coefficient, as in ordinary long division. The subtraction is performed by means of the feedback part of the shift register. Thus, the feedback shift register in Figure 7.9–1 performs division of two polynomials.

In our case, $g_{n-k} = g_0 = 1$, and for binary codes the arithmetic operations are performed in modulo-2 arithmetic. Consequently, the subtraction operations reduce to modulo-2 addition. Furthermore, we are interested only in generating the parity check

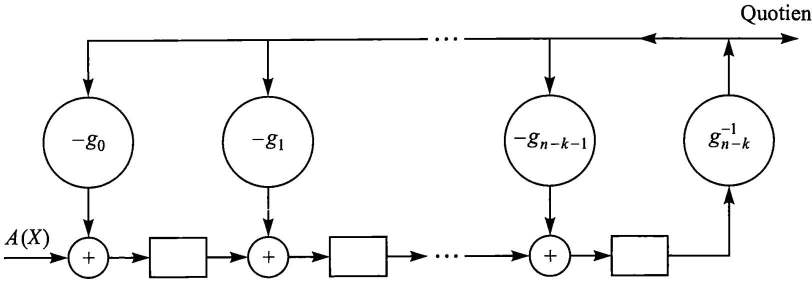


FIGURE 7.9–1
A feedback shift register for dividing the polynomial $A(X)$ by $g(X)$.

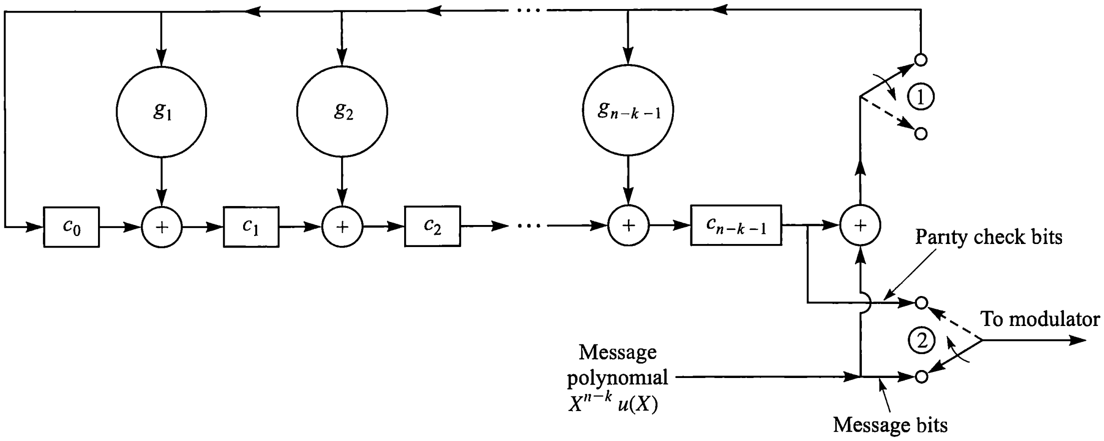


FIGURE 7.9–2
Encoding a cyclic code by use of the generator polynomial $g(X)$.

bits for each codeword, since the code is systematic. Consequently, the encoder for the cyclic code takes the form illustrated in Figure 7.9–2. The first k bits at the output of the encoder are simply the k information bits. These k bits are also clocked simultaneously into the shift register, since switch 1 is in the closed position. Note that the polynomial multiplication of X^{n-k} with $u(X)$ is not performed explicitly. After the k information bits are all clocked into the encoder, the positions of the two switches are reversed. At this time, the contents of the shift register are simply the $n - k$ parity check bits, which correspond to the coefficients of the remainder polynomial. These $n - k$ bits are clocked out one at a time and sent to the modulator.

EXAMPLE 7.9–7. The shift register for encoding the (7, 4) cyclic code with generator polynomial $g(X) = X^3 + X + 1$ is illustrated in Figure 7.9–3. Suppose the input message bits are 0110. The contents of the shift register are as follows:

Input	Shift	Shift Register Contents
	0	000
0	1	000
1	2	110
1	3	101
0	4	100

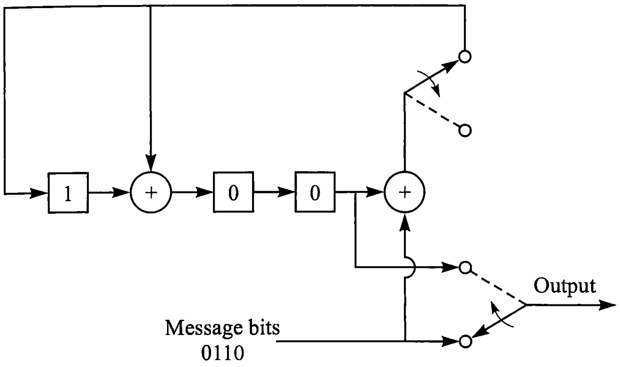


FIGURE 7.9–3
The encoder for the (7, 4) cyclic code with generator polynomial $g(X) = X^3 + X + 1$.

Hence, the three parity check bits are 100, which correspond to the code bits $c_5 = 0$, $c_6 = 0$, and $c_7 = 1$.

Instead of using the generator polynomial, we may implement the encoder for the cyclic code by making use of the parity polynomial

$$h(X) = X^k + h_{k-1}X^{k-1} + \cdots + h_1 X + 1$$

The encoder is shown in Figure 7.9–4. Initially, the k information bits are shifted into the shift register and simultaneously fed to the modulator. After all k information bits are in the shift register, the switch is thrown into position 2 and the shift register is clocked $n - k$ times to generate the $n - k$ parity check bits, as illustrated in Figure 7.9–4.

EXAMPLE 7.9–8. The parity polynomial for the (7, 4) cyclic code generated by $g(X) = X^3 + X + 1$ is $h(X) = X^4 + X^2 + X + 1$. The encoder for this code based on the parity polynomial is illustrated in Figure 7.9–5. If the input to the encoder is the message bits 0110, the parity check bits are $c_5 = 0$, $c_6 = 0$, and $c_7 = 1$, as is easily verified. Note that the encoder based on the generator polynomial is simpler when $n - k < k$ $\left(k > \frac{n}{2}\right)$, i.e., for high-rate codes $\left(R_c > \frac{1}{2}\right)$, while the encoder based on the parity polynomial is simpler when $k < n - k$ $\left(k < \frac{n}{2}\right)$, which corresponds to low- rate codes $\left(R_c < \frac{1}{2}\right)$.

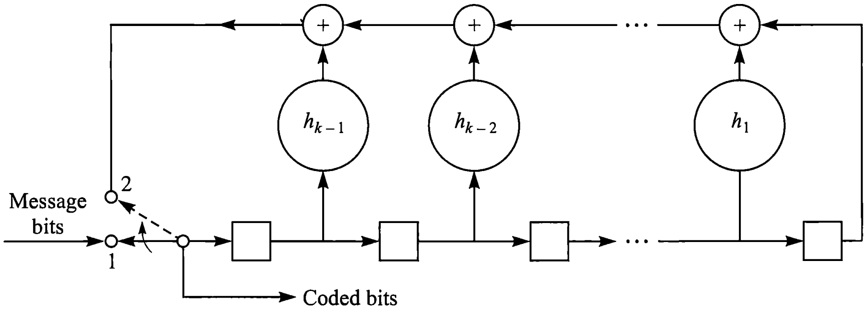


FIGURE 7.9–4
The encoder for an (n, k) cyclic code based on the parity polynomial $h(X)$.

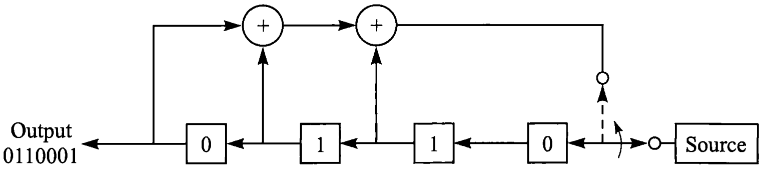


FIGURE 7.9–5
The encoder for the (7, 4) cyclic code based on the parity polynomial $h(X) = X^4 + X^2 + X + 1$.

7.9–4 Decoding Cyclic Codes

Syndrome decoding, described in Section 7.5, can be used for the decoding of cyclic codes. The cyclic structure of these codes makes it possible to implement syndrome computation and the decoding process using shift registers with considerable less complexity compared to the general class of linear block codes.

Let us assume that $\boldsymbol{c}$ is the transmitted codeword of a binary cyclic code and $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ is the received sequence at the output of the binary symmetric channel model (i.e., the channel output after the matched filter outputs have been passed through a binary quantizer). In terms of the corresponding polynomials, we can write

$$y(X) = c(X) + e(X) \tag{7.9–18}$$

and since $c(X)$ is a codeword, it is a multiple of $g(X)$, the generator polynomial of the code; i.e., $c(X) = u(X)g(X)$ for some $u(X)$, a polynomial of degree at most $k - 1$.

$$y(X) = u(X)g(X) + e(X) \tag{7.9–19}$$

From this relation we conclude

$$y(X) \mod g(X) = e(X) \mod g(X) \tag{7.9–20}$$

Let us define $s(X) = y(X) \mod g(X)$ to denote the remainder of dividing $y(X)$ by $g(X)$ and call $s(X)$ the *syndrome polynomial*, which is a polynomial of degree at most $n - k - 1$.

To compute the syndrome polynomial, we need to divide $y(X)$ by the generator polynomial $g(X)$ and find the remainder. Clearly $s(X)$ depends on the error pattern and not on the codeword, and different error patterns can yield the same syndrome polynomials since the number of possible syndrome polynomials is 2^{n-k} and the number of possible error patterns is 2^n. Maximum-likelihood decoding calls for finding the error pattern of the lowest weight corresponding to the computed syndrome polynomial $s(X)$ and adding it to $y(X)$ to obtain the most likely transmitted codeword polynomial $c(X)$.

The division of $y(X)$ by the generator polynomial $g(X)$ may be carried out by means of a shift register which performs division as described previously. First the received vector $\boldsymbol{y}$ is shifted into an $(n - k)$-stage shift register as illustrated in Figure 7.9–6. Initially, all the shift register contents are zero, and the switch is closed in position 1. After the entire n-bit received vector has been shifted into the register, the contents of the $n - k$ stages constitute the syndrome with the order of the bits numbered as shown in Figure 7.9–6. These bits may be clocked out by throwing the switch into

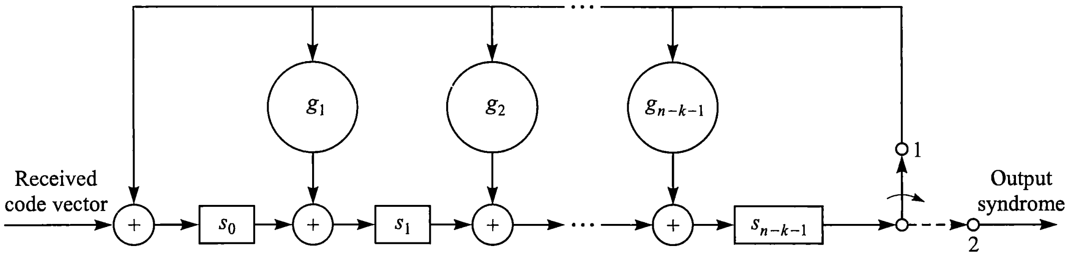


FIGURE 7.9–6
An $(n - k)$-stage shift register for computing the syndrome.

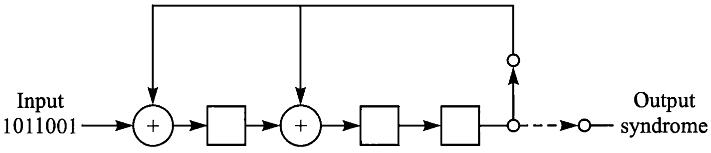
position 2. Given the syndrome from the $(n-k)$-stage shift register, a table lookup may be performed to identify the most probable error vector. Note that if the code is used for error detection, a nonzero syndrome detects an error in transmission of the codeword.

EXAMPLE 7.9–9. Let us consider the syndrome computation for the (7, 4) cyclic Hamming code generated by the polynomial $g(X) = X^3 + X + 1$. Suppose that the received vector is $\boldsymbol{y} = (1001101)$. This is fed into the three-stage register shown in Figure 7.9–7. After seven shifts, the contents of the shift register are 110, which corresponds to the syndrome $\boldsymbol{s} = (011)$. The most probable error vector corresponding to this syndrome is $\boldsymbol{e} = (0001000)$ and, hence,

$$\hat{\boldsymbol{c}} = \boldsymbol{y} + \boldsymbol{e} = (1000101)$$

The information bits are 1 0 0 0.

The table lookup decoding method using the syndrome is practical only when $n-k$ is small, e.g., when $n - k < 10$. This method is impractical for many interesting and powerful codes. For example, if $n-k = 20$, the table has 2^{20} (approximately 1 million)



Shift	Register contents
0	000
1	100
2	010
3	001
4	010
5	101
6	100
7	110

FIGURE 7.9–7
Syndrome computation for the (7, 4) cyclic code with generator polynomial $g(X) = X^3 + X + 1$ and received vector $\boldsymbol{y} = (1001101)$.

entries. Such a large amount of storage and the time required to locate an entry in such a large table renders the table lookup decoding method impractical for long codes having large numbers of check bits.

The cyclic structure of the code can be used to simplify finding the error polynomial. First we note that, as shown in Problem 7.54, if $s(X)$ is the syndrome corresponding to error sequence $e(X)$, then the syndrome corresponding to $e^{(1)}(X)$, the right cyclic shift of $e(X)$, is $s^{(1)}(X)$, defined by

$$s^{(1)}(X) = Xs(X) \mod g(X) \tag{7.9–21}$$

This means that to obtain the syndrome corresponding to $\boldsymbol{y}^{(1)}$, we need to multiply $s(X)$ by X and then divide by $g(X)$; but this is equivalent to shifting the content of the shift register shown in Figure 7.9–6 to the right when the input is disconnected. This means that the same combinatorial logic circuit that computes e_{n-1} from $\boldsymbol{s}$ can be used to compute e_{n-2} from a shifted version of $\boldsymbol{s}$, i.e., $\boldsymbol{s}^{(1)}$. The resulting decoder is known as the *Meggit decoder* (Meggitt (1961)).

The Meggit decoder feeds the received sequence $\boldsymbol{y}$ into the syndrome computing circuit to compute $s(X)$; the syndrome is fed into a combinatorial circuit that computes e_{n-1}. The output of this circuit is added modulo-2 to y_{n-1}, and after correction and a cyclic shift of the syndrome, the same combinatorial logic circuit computes e_{n-2}. This process is repeated n times, and if the error pattern is correctable, i.e., is one of the coset leaders, the decoder is capable of correcting it.

For details on the structure of decoders for general cyclic codes, the interested reader is referred to the texts of Peterson and Weldon (1972), Lin and Costello (2004), Blahut (2003), Wicker (1995), and Berlekamp (1968).

7.9–5 Examples of Cyclic Codes

In this section we discuss certain examples of cyclic codes. We have have selected the cyclic Hamming, Golay, and maximum-length codes discussed previously as general linear block codes. The most important class of cyclic codes, i.e., the BCH codes, is discussed in Section 7.10.

Cyclic Hamming Codes

The class of cyclic codes includes the *cyclic Hamming codes*, which have a block length $n = 2^m - 1$ and $n - k = m$ parity check bits, where m is any positive integer. The cyclic Hamming codes are equivalent to the Hamming codes described in Section 7.3–2.

Cyclic Golay Codes

The linear (23, 12) Golay code described in Section 7.3–6 can be generated as a cyclic code by means of the generator polynomial

$$g(X) = X^{11} + X^9 + X^7 + X^6 + X^5 + X + 1 \tag{7.9–22}$$

The codewords have a minimum distance $d_{\min} = 7$.

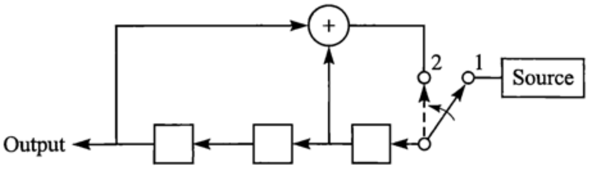


FIGURE 7.9–8
Three-stage ($m = 3$) shift register with feedback.

Maximum-Length Shift Register Codes

Maximum-length shift register codes are a class of cyclic codes equivalent to the maximum-length codes described in Section 7.3–3 as duals of Hamming codes. These are a class of cyclic codes with

$$(n, k) = (2^m - 1, m) \tag{7.9–23}$$

where m is a positive integer. The codewords are usually generated by means of an m-stage digital shift register with feedback, based on the parity polynomial. For each codeword to be transmitted, the m information bits are loaded into the shift register, and the switch is thrown from position 1 to position 2. The contents of the shift register are shifted to the left one bit at a time for a total of $2^m - 1$ shifts. This operation generates a systematic code with the desired output length $n = 2^m - 1$. For example, the codewords generated by the $m = 3$ stage shift register in Figure 7.9–8 are listed in Table 7.9–4.

Note that, with the exception of the all-zero codeword, all the codewords generated by the shift register are different cyclic shifts of a single codeword. The reason for this structure is easily seen from the state diagram of the shift register, which is illustrated in Figure 7.9–9 for $m = 3$. When the shift register is loaded initially and shifted $2^m - 1$ times, it will cycle through all possible $2^m - 1$ states. Hence, the shift register is back to its original state in $2^m - 1$ shifts. Consequently, the output sequence is periodic with length $n = 2^m - 1$. Since there are $2^m - 1$ possible states, this length corresponds to the largest possible period. This explains why the $2^m - 1$ codewords are different cyclic shifts of a single codeword. Maximum-length shift register codes exist for any positive

TABLE 7.9–4
Maximum-Length Shift Register Code for $m = 3$

Information Bits			Codewords						
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	1	0	1
0	1	0	0	1	0	0	1	1	1
0	1	1	0	1	1	1	0	1	0
1	0	0	1	0	0	1	1	1	0
1	0	1	1	0	1	0	0	1	1
1	1	0	1	1	0	1	0	0	1
1	1	1	1	1	1	0	1	0	0

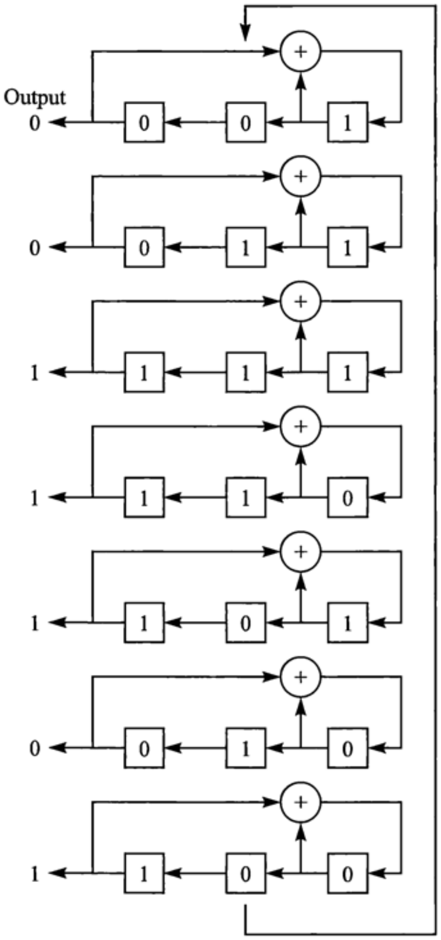


FIGURE 7.9–9
The seven states for the $m = 3$ maximum-length shift register.

value of m. Table 7.9–5 lists the stages connected to the modulo-2 adder that result in a maximum-length shift register for $2 \leq m \leq 34$.

Another characteristic of the codewords in a maximum-length shift register code is that each codeword, with the exception of the all-zero codeword, contains 2^{m-1} ones

TABLE 7.9–5
Shift-Register Connections for Generating Maximum-Length Sequences [from Forney (1970)].

m	Stages Connected to Modulo-2 Adder	m	Stages Connected to Modulo-2 Adder	m	Stages Connected to Modulo-2 Adder
2	1,2	13	1,10,11,13	24	1,18,23,24
3	1,3	14	1,5,9,14	25	1,23
4	1,4	15	1,15	26	1,21,25,26
5	1,4	16	1,5,14,16	27	1,23,26,27
6	1,6	17	1,15	28	1,26
7	1,7	18	1,12	29	1,28
8	1,5,6,7	19	1,15,18,19	30	1,8,29,30
9	1,6	20	1,18	31	1,29
10	1,8	21	1,20	32	1,11,31,32
11	1,10	22	1,22	33	1,21
12	1,7,9,12	23	1,19	34	1,8,33,34

and $2^{m-1}-1$ zeros, as shown in Problem 7.23. Hence all these codewords have identical weights, namely, $w = 2^{m-1}$. Since the code is linear, this weight is also the minimum distance of the code, i.e.,

$$d_{\min} = 2^{m-1}$$

As stated in Section 7.3–3, the maximum-length shift register code shown in Table 7.9–4 is identical to the $(7, 3)$ code given in Table 7.9–3, which is the dual of the $(7, 4)$ Hamming code given in Table 7.9–2. The maximum-length shift register codes are the dual codes of the cyclic Hamming $(2^m - 1, 2^m - 1 - m)$ codes. The shift register for generating the maximum-length code may also be used to generate a periodic binary sequence with period $n = 2^m - 1$. The binary periodic sequence exhibits a periodic autocorrelation $R(m)$ with values $R(m) = n$ for $m = 0, \pm n, \pm 2n, \ldots,$ and $R(m) = -1$ for all other shifts as described in Section 12.2–4. This impulselike autocorrelation implies that the power spectrum is nearly white, and hence the sequence resembles white noise. As a consequence, maximum-length sequences are called *pseudo-noise* (PN) sequences and find use in the scrambling of data and in the generation of spread spectrum signals as discussed in Chapter 12.

7.10
BOSE-CHAUDHURI-HOCQUENGHEM (BCH) CODES

BCH codes comprise a large class of cyclic codes that include codes over both binary and nonbinary alphabets. BCH codes have rich algebraic structure that makes their decoding possible by using efficient algebraic decoding algorithms. In addition, BCH codes exist for a wide range of design parameters (rates and block lengths) and are well tabulated. It also turns out that BCH codes are among the best-known codes for low to moderate block lengths.

Our study of BCH codes is rather brief, and the interested reader is referred to standard texts on coding theory including those by Wicker (1995), Lin and Costello (2004), Berlekamp (1968), and Peterson and Weldon (1972) for details and proofs.

7.10–1 The Structure of BCH Codes

BCH codes are a subclass of cyclic codes that were introduced independently by Bose Ray-Chaudhuri (1960a, 1960b) and Hocquenghem (1959). These codes have rich algebraic structure that makes it possible to design efficient algebraic decoding algorithms for them.

Since BCH codes are cyclic codes, we can describe them in terms of their generator polynomial $g(X)$. In this section we treat only a special class of binary BCH codes called *primitive binary BCH codes*. These codes have a block length of $n = 2^m - 1$ for some integer $m \geq 3$, and they can be designed to have a guaranteed error detection capability of at least t errors for any $t < 2^{m-1}$. In fact for any two positive integers $m \geq 3$ and $t < 2^{m-1}$ we can design a BCH code whose parameters satisfy the following relations:

$$\begin{aligned} n &= 2^m - 1 \\ n - k &\leq mt \\ d_{\min} &\geq 2t + 1 \end{aligned} \tag{7.10–1}$$

The first equality determines the block length of the code. The second inequality provides a bound on the number of parity check bits of the code, and the third inequality states that this code is capable of correcting at least t errors. The resulting code is called a t-error correcting BCH code; although it is possible that this code can correct more than t errors.

The Generator Polynomial for BCH Codes

To design a t-error correcting (primitive) BCH code, we choose α, a primitive element of GF(2^m). Then $g(X)$, the generator polynomial of the BCH code, is defined as the lowest-degree polynomial $g(X)$ over GF(2) such that α, α^2, $\alpha^3, \ldots,$ and α^{2t} are its roots.

Using the definition of the minimal polynomial of a field element given in Section 7.1–1 and by Equation 7.1–12, we know that any polynomial over GF(2) that has $\beta \in$ GF(2) as a root is divisible by $\phi_\beta(X)$, the minimal polynomial of β. Therefore $g(X)$ must be divisible by $\phi_{\alpha^i}(X)$ for $1 \leq i \leq 2t$. Since $g(X)$ is a polynomial of lowest degree with this property, we conclude that

$$g(X) = \text{LCM}\,\{\phi_{\alpha^i}(X), 1 \leq i \leq 2t\} \tag{7.10–2}$$

where LCM denotes the least common multiple of $\phi_{\alpha^i}(X)$'s. Also note that, for instance, the $\phi_{\alpha^i}(X)$ for $i = 1, 2, 4, \ldots$ are the same since $\alpha, \alpha^2, \alpha^4, \ldots$ are conjugates and hence they have the same minimal polynomial. The same is true for $\alpha^3, \alpha^6, \alpha^{12}, \ldots$. Therefore, in the expression for $g(X)$ it is sufficient to consider only odd values of α, i.e.,

$$g(X) = \text{LCM}\,\{\phi_\alpha(X), \phi_{\alpha^3}(X), \phi_{\alpha^5}(X), \ldots, \phi_{\alpha^{2t-1}}(X)\} \tag{7.10–3}$$

and since the degree of $\phi_{\alpha^i}(X)$ does not exceed m, the degree of $g(X)$ is at most mt. Therefore, $n - k \leq mt$.

Let us assume that $c(X)$ is a codeword polynomial of the designed BCH code. From the cyclic property of the code we know that $g(X)$ is a divisor of $c(X)$. Therefore, all α^i for $1 \leq i \leq 2t$ are roots of $c(X)$; i.e., for any codeword polynomial $c(X)$ we have

$$c\left(\alpha^i\right) = 0 \qquad 1 \leq i \leq 2t \tag{7.10–4}$$

The conditions given in Equation 7.10–4 are necessary and sufficient conditions for a polynomial of degree less than n to be a codeword polynomial of the BCH code.

EXAMPLE 7.10–1. To design a single-error-correcting ($t = 1$) BCH code with block length $n = 15$ ($m = 4$), we choose α a primitive element in GF(2^4). The minimal polynomial of α is a primitive polynomial of degree 4.

From Table 7.1–5 we see that $g(X) = \phi_\alpha(X) = X^4 + X + 1$. Therefore, $n - k = 4$ and $k = 11$. Since the weight of $g(X)$ is 3, we have $d_{\min} \geq 3$. Combining this with Equation 7.10–1, which states $d_{\min} \leq 2t + 1 = 3$, we conclude that $d_{\min} = 3$. Therefore a single-error-correcting BCH code with block length 15 is a (15, 11) code with $d_{\min} = 3$. This is, in fact, a cyclic Hamming code. In general, cyclic Hamming codes are single-error-correcting BCH codes.

EXAMPLE 7.10–2. To design a four-error-correcting ($t = 4$) BCH code with block length $n = 15$ ($m = 4$), we choose α a primitive element in GF(2^4). The minimal polynomial of α is $g(X) = \phi_\alpha(X) = X^4 + X + 1$. We also need to find the minimal polynomials of α^3, α^5, and α^7.

From Example 7.1–5 we have $\phi_{\alpha^3} = X^4 + X^3 + X^2 + X + 1$, $\phi_{\alpha^5} = X^2 + X + 1$, and $\phi_{\alpha^7}(X) = X^4 + X^3 + 1$. Therefore,

$$\begin{aligned} g(X) &= (X^4 + X + 1)(X^4 + X^3 + X^2 + X + 1) \\ &\quad \times (X^2 + X + 1)(X^4 + X^3 + 1) \\ &= X^{14} + X^{13} + X^{12} + X^{11} + X^{10} + X^9 + X^8 + X^7 \\ &\quad + X^6 + X^5 + X^4 + X^3 + X^2 + X + 1 \end{aligned} \tag{7.10–5}$$

Hence $n - k = 14$ and $k = 1$; the resulting code is a (15, 1) repetition code with $d_{\min} = 15$. Note that this code was designed to correct four errors but it is capable of correcting up to seven errors.

EXAMPLE 7.10–3. To design a double-error-correcting BCH code with block length $n = 15$ ($m = 4$), we need the minimal polynomials of α and α^3. The minimal polynomial of α is $g(X) = \phi_\alpha(X) = X^4 + X + 1$, and from Example 7.1–5, $\phi_{\alpha^3} = X^4 + X^3 + X^2 + X + 1$. Therefore,

$$\begin{aligned} g(X) &= (X^4 + X + 1)(X^4 + X^3 + X^2 + X + 1) \\ &= X^8 + X^7 + X^6 + X^4 + 1 \end{aligned} \tag{7.10–6}$$

Hence $n - k = 8$ and $k = 7$, and the resulting code is a (15, 7) BCH code with $d_{\min} = 5$.

Table 7.10–1 lists the coefficients of generator polynomials for BCH codes of block lengths $7 \leq n \leq 255$, corresponding to $3 \leq m \leq 8$. The coefficients are given in octal form, with the leftmost digit corresponding to the highest-degree term of the generator polynomial. Thus, the coefficients of the generator polynomial for the (15, 5) code are 2467, which in binary form is 10100110111. Consequently, the generator polynomial is $g(X) = X^{10} + X^8 + X^5 + X^4 + X^2 + X + 1$. A more extensive list of generator polynomials for BCH codes is given by Peterson and Weldon (1972), who tabulated the polynomial factors of $X^{2m-1} + 1$ for $m \leq 34$.

Let us consider from Table 7.10–1 the sequence of BCH codes with triplet parameters (n, k, t) such that for these codes R_c is close to $\frac{1}{2}$. These codes include (7, 4, 1), (15, 8, 2), (31, 16, 3), (63, 30, 6), (127, 64, 10), and (255, 131, 18) codes. We observe that as n increases and the rate remains almost constant, the ratio $\frac{t}{n}$, that is the fraction of errors that the code can correct, decreases. In fact for all BCH codes with constant rate, as the block length increases, the fraction of correctable errors goes to zero. This shows that the BCH codes are asymptotically bad, and for large n their δ_n falls below

TABLE 7.10–1
Coefficients of Generator Polynomials (in Octal Form) for BCH Codes of Length $7 \le n \le 255$

n	k	t	$g(X)$
7	4	1	13
15	11	1	23
	7	2	721
	5	3	2467
31	26	1	45
	21	2	3551
	16	3	107657
	11	5	5423325
	6	7	313365047
63	57	1	103
	51	2	12471
	45	3	1701317
	39	4	166623567
	36	5	1033500423
	30	6	157464165547
	24	7	17323260404441
	18	10	1363026512351725
	16	11	6331141367235453
	10	13	472622305527250155
	7	15	5231045543503271737
127	120	1	211
	113	2	41567
	106	3	11554743
	99	4	3447023271
	92	5	624730022327
	85	6	130704476322273
	78	7	26230002166130115
	71	9	6255010713253127753
	64	10	1206534025570773100045
	57	11	335265252057050535177 21
	50	13	54446512523314012421501421
	43	14	17721772213651227521220574343
	36	15	3146074666522075044764574721735
	29	21	403114461367670603667530141176155
	22	23	123376070404722522435445626637647043
	15	27	22057042445604554770523013762217604353
	8	31	7047264052751030651476224271567733130217
255	247	1	435
	239	2	267543
	231	3	156720665
	223	4	75626641375
	215	5	23157564726421
	207	6	16176560567636227
	199	7	7633031270420722341
	191	8	2663470176115333714567
	187	9	52755313540001322236351
	179	10	22624710717340432416300455
	171	11	15416214212342356077061630 67

(continued)

TABLE 7.10–1
(Continued)

n	*k*	*t*	*g*(*X*)
	163	12	7500415510075602551574724514601
	155	13	3757513005407665015722506464677633
	147	14	1642130173537165525304165305441011711
	139	15	461401732060175561570722730247453567445
	131	18	215713331471510151261250277442142024165471
	123	19	120614052242066003717210326516141226272506267
	115	21	60526665572100247263636404600276352556313472737
	107	22	22205772322066256312417300235347420176574750154441
	99	23	10656667253473174222741416201574332252411076432303431
	91	25	6750265030327444172723631724732511075550762720724344561
	87	26	110136763414743236435231634307172046206722545273311721317
	79	27	66700035637657500020270344207366174621015326711766541342355
	71	29	24024710520644321515554172112331163205444250362557643221706035
	63	30	10754475055163544325315217357707003666111726455267613656702543301
	55	31	7315425203501100133015275306032054325414326755010557044426035473617
	47	42	2533542017062646563033041377406233175123334145446045005066024552543173
	45	43	15202056055234161131101346376423701563670024470762373033202157025051541
	37	45	5136330255067007414177447447245437530420735706174323432347644354737403044003
	29	47	3025715536673071465527064012361377115342242324201174114060254757410403565037
	21	55	1256215257060332656001773153607612103227341405653074542521153121614466513473725
	13	59	464173200505256454442657371425006600433067744547656140317467721357026134460500547
	9	63	15726025217472463201031043255355134614162367212044074545112766115547705561677516057

the Varshamov-Gilbert bound. We need, however, to keep in mind that this happens at large values of n and for small to moderate values of n, which include the most practical cases, these codes remain among the best-known codes for which efficient decoding algorithms are known.

7.10–2 Decoding BCH Codes

Since BCH codes are cyclic codes, any decoding algorithm for cyclic codes can be applied to BCH codes. For instance, BCH codes can be decoded using a Meggit decoder. However, the additional structure in BCH codes makes it possible to use more efficient decoding algorithms, particularly when using codes with long block lengths.

Let us assume that a codeword $\boldsymbol{c}$ is associated with codeword polynomial $c(X)$. By Equation 7.10–4, we know that $c(\alpha^i) = 0$ for $1 \le i \le 2t$. Let us assume that the error polynomial is $e(X)$ and the received polynomial is $y(X)$. Then

$$y(X) = c(X) + e(X) \qquad (7.10\text{–}7)$$

Let us denote the value of $y(X)$ at α^i by S_i, i.e., the *syndromes* defined by

$$\begin{aligned} S_i &= y(\alpha^i) \\ &= c(\alpha^i) + e(\alpha^i) \qquad 1 \le i \le 2t \qquad (7.10\text{–}8) \\ &= e(\alpha^i) \end{aligned}$$

Obviously if $e(X)$ is zero, or it is equal to a nonzero codeword, the syndromes are all zero. The syndrome can be computed from the received sequence $\boldsymbol{y}$ using GF(2^m) arithmetic.

Now let us assume there have been ν errors in transmission of $\boldsymbol{c}$, where $\nu \leq t$. Let us denote the location of these errors by $j_1, j_2, \ldots, j_\nu$, where without loss of generality we may assume $0 \leq j_1 < j_2 < \cdots < j_\nu \leq n-1$. Therefore

$$e(X) = X^{j_\nu} + X^{j_{\nu-1}} + \cdots + X^{j_2} + X^{j_1} \tag{7.10–9}$$

From Equations 7.10–8 and 7.10–9 we conclude that

$$\begin{aligned} S_1 &= \alpha^{j_1} + \alpha^{j_2} + \cdots + \alpha^{j_\nu} \\ S_2 &= (\alpha^{j_1})^2 + (\alpha^{j_2})^2 + \cdots + (\alpha^{j_\nu})^2 \\ &\vdots \\ S_{2t} &= (\alpha^{j_1})^{2t} + (\alpha^{j_2})^{2t} + \cdots + (\alpha^{j_\nu})^{2t} \end{aligned} \tag{7.10–10}$$

These are a set of $2t$ equations in ν unknowns, namely, $j_1, j_2, \ldots, j_\nu$, or equivalently α^{j_i}, $1 \leq i \leq \nu$. Any method for solving simultaneous equations can be applied to find unknowns α^{j_i} from which error locations $j_1, j_2, \ldots, j_\nu$ can be found. Having determined error locations, we change the received bit at those locations to find the transmitted codeword $\boldsymbol{c}$.

By defining *error location numbers* $\beta_i = \alpha^{j_i}$ for $1 \leq i \leq \nu$, Equation 7.10–10 becomes

$$\begin{aligned} S_1 &= \beta_1 + \beta_2 + \cdots + \beta_\nu \\ S_2 &= \beta_1^2 + \beta_2^2 + \cdots + \beta_\nu^2 \\ &\vdots \\ S_{2t} &= \beta_1^{2t} + \beta_2^{2t} + \cdots + \beta_\nu^{2t} \end{aligned} \tag{7.10–11}$$

Solving this set of equations determines β_i for $1 \leq i \leq \nu$ from which error locations can be determined. Obviously the β_i's are members of GF(2^m), and solving these equations requires arithmetic over GF(2^m). This set of equations in general has many solutions. For maximum-likelihood (minimum Hamming distance) decoding we are interested in a solution with the smallest number of β's.

To solve these equations, we introduce the *error locator polynomial* as

$$\begin{aligned} \sigma(X) &= (1 + \beta_1 X)(1 + \beta_2 X) \cdots (1 + \beta_\nu X) \\ &= \sigma_\nu X^\nu + \sigma_{\nu-1} X^{\nu-1} + \cdots + \sigma_1 X + \sigma_0 \end{aligned} \tag{7.10–12}$$

whose roots are β_i^{-1} for $1 \leq i \leq \nu$. Finding the roots of this polynomial determines the location of errors. We need to determine σ_i for $0 \leq i \leq \nu$ to have $\sigma(X)$ from which we can find the roots and hence locate the errors. Expanding Equation 7.10–12 results in the following set of equations:

$$\begin{aligned}\sigma_0 &= 1\\ \sigma_1 &= \beta_1 + \beta_2 + \cdots + \beta_\nu\\ \sigma_2 &= \beta_1\beta_2 + \beta_1\beta_3 + \cdots + \beta_{\nu-1}\beta_\nu\\ &\vdots\\ \sigma_\nu &= \beta_1\beta_2\cdots\beta_\nu\end{aligned} \tag{7.10–13}$$

Using Equations 7.10–10 and 7.10–13, we obtain the following set of equations relating the coefficients of $\sigma(X)$ and the syndromes.

$$\begin{aligned} S_1 + \sigma_1 &= 0\\ S_2 + \sigma_1 S_1 + 2\sigma_2 &= 0\\ S_3 + \sigma_1 S_2 + \sigma_2 S_1 + 3\sigma_3 &= 0\\ &\vdots\\ S_\nu + \sigma_1 S_{\nu-1} + \cdots + \sigma_{\nu-1} S_1 + \nu\sigma_\nu &= 0\\ S_{\nu+1} + \sigma_1 S_\nu + \cdots + \sigma_{\nu-1} S_2 + \sigma_\nu S_1 &= 0\\ &\vdots \end{aligned} \tag{7.10–14}$$

We need to obtain the lowest-degree polynomial $\sigma(X)$ whose coefficients satisfy this set of equations. After determining $\sigma(X)$, we have to find its roots β_i^{-1}. The inverse of the roots provides the location of the errors. Note that when the polynomial of the lowest degree $\sigma(X)$ is found, we can simply find its roots over GF(2^m) by substituting the 2^m field elements in the polynomial.

The Berlekamp-Massey Decoding Algorithm for BCH Codes

Several algorithms have been proposed for solution of Equation 7.10–14. Here we present the well-known Berlekamp-Massey algorithm due to Berlekamp (1968) and Massey (1969). Our presentation of this algorithm follows the presentation in Lin and Costello (2004). The interested reader is referred to Lin and Costello (2004), Berlekamp (1968), Peterson and Weldon (1972), MacWilliams and Sloane (1977), Blahut (2003), or Wicker (1995) for details and proofs.

To implement the Berlekamp-Massey algorithm, we begin by finding a polynomial of lowest degree $\sigma^{(1)}(X)$ that satisfies the first equation in 7.10–14. In the second step we test to see if $\sigma^{(1)}(X)$ satisfies the second equation in 7.10–14. If it satisfies the second equation, we set $\sigma^{(2)}(X) = \sigma^{(1)}(X)$. Otherwise, we introduce a correction term to $\sigma^{(1)}(X)$ to obtain $\sigma^{(2)}(X)$, the polynomial of the lowest degree that satisfies the first two equations. This process is continued until we obtain a polynomial of minimum degree that satisfies all equations.

In general, if

$$\sigma^{(\mu)}(X) = \sigma_{l_\mu}^{(\mu)} X^{l_\mu} + \sigma_{l_{\mu-1}}^{(\mu)} X^{l_\mu} + \cdots + \sigma_2^{(\mu)} X^2 + \sigma_1^{(\mu)} X + 1 \tag{7.10–15}$$

is the polynomial of the lowest degree that satisfies the first μ equations in Equation 7.10–14, to find $\sigma^{(\mu+1)}(X)$ we compute the μth *discrepancy*, denoted by d_μ and given by

$$d_\mu = S_{\mu+1} + \sigma_1^{(\mu)} S_\mu + \sigma_2^{(\mu)} S_{\mu-1} + \cdots + \sigma_{l_\mu}^{(\mu)} S_{\mu+1-l_\mu} \tag{7.10–16}$$

If $d_\mu = 0$, no correction is necessary and the $\sigma^{(\mu)}(X)$ that satisfies the $(\mu+1)$st equation is Equation 7.10–14. In this case we set

$$\sigma^{(\mu+1)}(X) = \sigma^{(\mu)}(X) \tag{7.10–17}$$

If $d_\mu \neq 0$, a correction is necessary. In this case $\sigma^{(\mu+1)}(X)$ is given by

$$\sigma^{(\mu+1)}(X) = \sigma^{(\mu)}(X) + d_\mu d_\rho^{-1} \sigma^{(\rho)}(X) X^{\mu-\rho} \tag{7.10–18}$$

where $\rho < \mu$ is selected such that $d_\rho \neq 0$ and among all such ρ's the value of $\rho - l_\rho$ is maximum (l_ρ is the degree of $\sigma^{(\rho)}(X)$).

The polynomial given by Equation 7.10–18 is the polynomial of the lowest degree that satisfies the first $(\mu+1)$ equations in Equation 7.10–14. This process is continued until $\sigma^{(2t)}(X)$ is derived. The degree of this polynomial determines the number of errors, and its roots can be used to locate the errors, as explained earlier. If the degree of $\sigma^{(2t)}(X)$ is higher than t, the number of errors in the received sequence is greater than t, and the errors cannot be corrected.

The Berlekamp-Massey algorithm can be better carried out if we begin with a table such as Table 7.10–2.

EXAMPLE 7.10–4. Let us assume that the double-error-correcting BCH code designed in Example 7.10–3 is considered, and the binary received sequence at the output of the BSC channel is

$$y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1)$$

TABLE 7.10–2
The Berlekamp-Massey Algorithm

μ	$\sigma^{(\mu)}(X)$	d_μ	l_μ	$\mu - l_\mu$
-1	1	1	0	-1
0	1	S_1	0	0
1	$1 + S_1 X$			
2				
⋮				
$2t$				

TABLE 7.10–3
The Berlekamp-Massey Algorithm Implementation for Example 7.10–4

μ	$\sigma^{(\mu)}(X)$	d_μ	l_μ	$\mu - l_\mu$
-1	1	1	0	-1
0	1	α^{14}	0	0
1	$1+\alpha^{14}X$	0	1	0
2	$1+\alpha^{14}X$	α^2	1	1
3	$1+\alpha^{14}X+\alpha^3X^2$	0	2	1
4	$1+\alpha^{14}X+\alpha^3X^2$		2	2

The corresponding received polynomial is $y(X) = X^3 + 1$, and the syndrome computation yields

$$\begin{aligned} S_1 &= \alpha^3 + 1 = \alpha^{14} \\ S_2 &= \alpha^6 + 1 = \alpha^{13} \\ S_3 &= \alpha^9 + 1 = \alpha^7 \\ S_4 &= \alpha^{12} + 1 = \alpha^{11} \end{aligned} \tag{7.10–19}$$

where we have used Table 7.1–6. Now we have all we need to fill in the entries of Table 7.10–2 by using Equations 7.10–16 to 7.10–18. The result is given in Table 7.10–3.

Therefore $\sigma(X) = 1 + \alpha^{14}X + \alpha^3X^2$, and since the degree of this polynomial is 2, this corresponds to a correctable error pattern. We can find the roots of $\sigma(X)$ by inspection, i.e., by substituting the elements of GF(2^4). This will give the two roots of 1 and α^{12}. Since the roots are the reciprocals of the error location numbers, we conclude that the error location numbers are $\beta_1 = \alpha^0$ and $\beta_2 = \alpha^3$. From this the errors are at locations $j_1 = 0$ and $j_2 = 3$. From Equation 7.10–9 the error polynomial is $e(X) = 1 + X^3$, and $c(X) = y(X) + e(X) = 0$, i.e., the detected codeword, is the all-zero codeword.

7.11

REED-SOLOMON CODES

Reed-Solomon (RS) codes are probably the most widely used codes in practice. These codes are used in communication systems and particularly data storage systems. Reed-Solomon codes are a special class of nonbinary BCH codes that were first introduced in Reed and Solomon (1960). As we have already seen, these codes achieve the Singleton bound and hence belong to the class of MDS codes.

Recall that in construction of a binary BCH code of block length $n = 2^m - 1$, we began by selecting a primitive element in GF(2^m) and then finding the minimal polynomials of α^i for $1 \le i \le 2t$. The notion of the minimal polynomial as defined in Section 7.1–1 was a special case of the general notion of minimal polynomial with respect to a subfield. We defined the minimal of $\beta \in$ GF(2^m) as a polynomial of lowest degree over GF(2), where one of its roots is β. This is the definition of the minimal polynomial with respect to GF(2). If we drop the restriction that the minimal polynomial be defined over GF(2), we can have other minimal polynomials of lower degree. One extreme case occurs when we define the minimal polynomial of $\beta \in \mathrm{GF}(2^m)$ with respect to $\mathrm{GF}(2^m)$. In this case we look for a polynomial of lowest degree over $\mathrm{GF}(2^m)$ whose root is β. Obviously $X + \beta$ is such a polynomial.

Reed-Solomon codes are t-error-correcting 2^m-ary BCH codes with block length $N = 2^m - 1$ symbols (i.e., mN binary digits)[†]. To design a Reed-Solomon code, we choose $\alpha \in \mathrm{GF}(2^m)$ to be a primitive element and find the minimal polynomials of α^i, for $1 \leq i \leq 2t$, over $\mathrm{GF}(2^m)$. These polynomials are obviously of the form $X + \alpha^i$. Hence, the generator polynomial $g(X)$ is given by

$$\begin{aligned} g(X) &= (X+\alpha)(X+\alpha^2)(X+\alpha^3)\cdots(X+\alpha^{2t}) \\ &= X^{2t} + g_{2t-1}X^{2t-1} + \cdots + g_1 X + g_0 \end{aligned} \tag{7.11–1}$$

where $g_i \in \mathrm{GF}(2^m)$ for $0 \leq i \leq 2t-1$; i.e., $g(X)$ is a polynomial over $\mathrm{GF}(2^m)$. Since α^i, for $1 \leq i \leq 2t$, are nonzero elements of $\mathrm{GF}(2^m)$, they are all roots of $X^{2^m-1}+1$; therefore $g(X)$ is a divisor of $X^{2^m-1}+1$, and it is the generator polynomial of a 2^m-ary code with block length $N = 2^m - 1$ and $N - K = 2t$. Note that the weight of $g(X)$ cannot be less than $D_{\min}$, the minimum distance of the code, which is, by Equation 7.10–1, at least $2t+1$. This means that none of the g_i's in Equation 7.11–1 can be zero, and therefore the minimum weight of the resulting code *is equal to* $2t+1$. Therefore, for this code

$$D_{\min} = 2t + 1 = N - K + 1 \tag{7.11–2}$$

which shows that the code is MDS.

From the discussion above, we conclude that Reed-Solomon codes are 2^m-ary $(2^m-1, 2^m-2t-1)$ BCH codes with minimum distance $D_{\min} = 2t+1$, where m is any positive integer greater than or equal to 3 and $1 \leq t \leq 2^{m-1}-1$. Equivalently, we can define Reed-Solomon codes in terms of m and $D_{\min}$, the minimum distance of the code, as 2^m-ary BCH codes with $N = 2^m - 1$ and $K = N - D_{\min}$, where $3 \leq D_{\min} \leq n$.

EXAMPLE 7.11-1. To design a triple-error-correcting Reed-Solomon code of length $n = 15$, we note that $N = 15 = 2^4 - 1$. Therefore, $m = 4$ and $t = 3$. We choose $\alpha \in \mathrm{GF}(2^4)$ to be a primitive element. Using Equation 7.11–1, we obtain

$$\begin{aligned} g(X) &= (X+\alpha)(X+\alpha^2)(X+\alpha^3)(X+\alpha^4)(X+\alpha^5)(X+\alpha^6) \\ &= X^6 + \alpha^{10}X^5 + \alpha^{14}X^4 + \alpha^4X^3 + \alpha^6X^2 + \alpha^9X + \alpha^6 \end{aligned} \tag{7.11–3}$$

This is a (15, 8) triple-error-correcting Reed-Solomon code over $\mathrm{GF}(2^4)$. Codewords of this code have a block length of 15 where each component is a 2^4-ary symbol. In binary representation the codewords have length 60.

A popular Reed-Solomon code is the (255, 223) code over $\mathrm{GF}(2^8)$. This code has a minimum distance of $D_{\min} = 255 - 223 + 1 = 33$ and is capable of correcting 16 symbol errors. If these errors are spread, in the worst possible scenario this code is capable of correcting 16 bit errors. On the other hand, if these errors occur as a cluster, i.e., if we have a burst of errors, this code can correct any burst of length $14 \times 8 + 2 = 114$ bits. Some bursts of length up to $16 \times 8 = 128$ errors can be corrected also by this code. That is the reason why Reed-Solomon codes are particularly attractive in channels with burst of errors. Such channels include fading channels and storage channels in which scratches and manufacturing imperfections usually damage a sequence of bits. Reed-Solomon codes are also popular in concatenated coding schemes discussed later in this chapter.

[†] In general, RS codes are defined on $\mathrm{GF}(p^m)$. For Reed-Solomon codes we denote the block length by N (symbols) and the number of information symbols by K. The minimum distance is denoted by $D_{\min}$.

Since Reed-Solomon codes are BCH codes, any algorithm used for decoding BCH codes can be used for decoding Reed-Solomon codes. The Berlekamp-Massey algorithm, for instance, can be used for the decoding of Reed-Solomon codes. The only difference is that after locating the errors, we also have to determine the values of the errors. This step was not necessary in binary BCH codes since in that case the value of any error is 1 that changes a 0 to a 1 and a 1 to a 0. In nonbinary BCH codes that is not the case. The value of error can be any nonzero member of GF(2^m) and has to be determined. The methods used to determine the value of errors are beyond the scope of our treatment. The interested user is referred to Lin and Costello (2004).

An interesting property of Reed-Solomon codes is that their weight enumeration polynomial is known. In general, the weight distribution of a Reed-Solomon code with symbols from GF(q) and with block length $N = q - 1$ and minimum distance $D_{\min}$ is given by

$$A_i = \binom{N}{i} N \sum_{j=0}^{i-D_{\min}} (-1)^j \binom{i-1}{j} (N+1)^{i-j-D_{\min}}, \qquad \text{for } D_{\min} \le i \le N \quad (7.11\text{–}4)$$

A nonbinary code is particularly matched to an M-ary modulation technique for transmitting the 2^m possible symbols. Specifically, M-ary orthogonal signaling, e.g., M-ary FSK, is frequently used. Each of the 2^m symbols in the 2^m-ary alphabet is mapped to one of the $M = 2^m$ orthogonal signals. Thus, the transmission of a codeword is accomplished by transmitting N orthogonal signals, where each signal is selected from the set of $M = 2^m$ possible signals.

The optimum demodulator for such a signal corrupted by AWGN consists of M matched filters (or cross-correlators) whose outputs are passed to the decoder, either in the form of soft decisions or in the form of hard decisions. If hard decisions are made by the demodulator, the symbol error probability P_M and the code parameters are sufficient to characterize the performance of the decoder. In fact, the modulator, the AWGN channel, and the demodulator form an equivalent discrete (M-ary) input, discrete (M-ary) output, symmetric memoryless channel characterized by the transition probabilities $P_c = 1 - P_M$ and $P_M/(M-1)$. This channel model, which is illustrated in Figure 7.11–1, is a generalization of the BSC.

The performance of the hard decision decoder may be characterized by the following upper bound on the codeword error probability:

$$P_e \le \sum_{i=t+1}^{N} \binom{N}{i} P_M^i (1 - P_M)^{N-i} \qquad (7.11\text{–}5)$$

where t is the number of errors guaranteed to be corrected by the code.

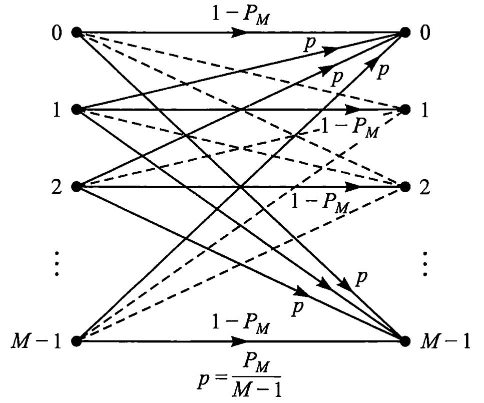


FIGURE 7.11–1
An M-ary input, M-ary output, symmetric memoryless channel.

When a codeword error is made, the corresponding symbol error probability is

$$P_{es} = \frac{1}{N} \sum_{i=t+1}^{N} i \binom{N}{i} P_M^i (1 - P_M)^{N-i} \tag{7.11–6}$$

Furthermore, if the symbols are converted to binary digits, the bit error probability corresponding to Equation 7.11–6 is

$$P_{eb} = \frac{2^{m-1}}{2^m - 1} P_{es} \tag{7.11–7}$$

EXAMPLE 7.11–2. Let us evaluate the performance of an $N = 2^5 - 1 = 31$ Reed-Solomon code with $D_{\min} = 3, 5, 9$, and 17. The corresponding values of K are 29, 27, 23, and 15. The modulation is $M = 32$ orthogonal FSK with noncoherent detection at the receiver. The probability of a symbol error is given by Equation 4.5–44 and may be expressed as

$$P_e = \frac{1}{M} e^{-\gamma} \sum_{i=2}^{M} (-1)^i \binom{M}{i} e^{\gamma/i} \tag{7.11–8}$$

where γ is the SNR per code symbol. By using Equation 7.11–8 in Equation 7.11–6 and combining the result with Equation 7.11–7, we obtain the bit error probability. The results of these computations are plotted in Figure 7.11–2. Note that the more powerful codes (large $D_{\min}$) give poorer performance at low SNR per bit than the weaker codes. On the other hand, at high SNR, the more powerful codes give better performance. Hence, there are crossovers among the various codes, as illustrated, for example, in Figure 7.11–2 for the $t = 1$ and $t = 8$ codes. Crossovers also occur among the $t = 1, 2$, and 4 codes at smaller values of SNR per bit. Similarly, the curves for $t = 4$ and 8 and for $t = 8$ and 2 cross in the region of high SNR. This is the characteristic behavior for noncoherent detection of the coded waveforms.

If the demodulator does not make a hard decision on each symbol, but instead passes the unquantized matched filter outputs to the decoder, soft decision decoding can be performed. This decoding involves the formation of $q^K = 2^{mK}$ correlation metrics, where each metric corresponds to one of the q^K codewords and consists of a sum of N matched filter outputs corresponding to the N code symbols. The matched filter outputs may be added coherently, or they may be envelope-detected and then

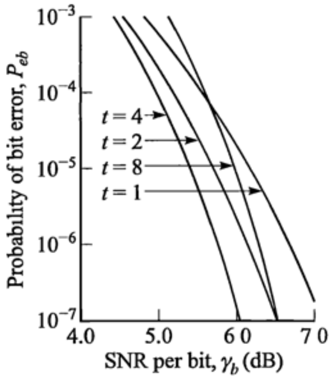


FIGURE 7.11–2
Performance of several $N = 31$, t-error-correcting Reed-Solomon codes with 32-ary FSK modulation on an AWGN channel (noncoherent demodulation)

added, or they may be square-law-detected and then added. If coherent detection is used and the channel noise is AWGN, the computation of the probability of error is a straightforward extension of the binary case considered in Section 7.4. On the other hand, when envelope detection or square-law detection and noncoherent combining are used to form the decision variables, the computation of the decoder performance is considerably more complicated.

7.12
CODING FOR CHANNELS WITH BURST ERRORS

Most of the well-known codes that have been devised for increasing reliability in the transmission of information are effective when the errors caused by the channel are statistically independent. This is the case for the AWGN channel. However, there are channels that exhibit bursty error characteristics. One example is the class of channels characterized by multipath and fading, which is described in detail in Chapter 13. Signal fading due to time-variant multipath propagation often causes the signal to fall below the noise level, thus resulting in a large number of errors. A second example is the class of magnetic recording channels (tape or disk) in which defects in the recording media result in clusters of errors. Such error clusters are not usually corrected by codes that are optimally designed for statistically independent errors.

Some of the codes designed for random error correction, i.e., nonburst errors, have the capability of burst error correction. A notable example is Reed-Solomon codes that can easily correct long burst of errors because such long error bursts result in a few symbol errors that can be easily corrected. Considerable work has been done on the construction of codes that are capable of correcting burst errors. Probably the best-known burst error correcting codes are the subclass of cyclic codes called Fire codes, named after P. Fire (Fire (1959)), who discovered them. Another class of cyclic codes for burst error correction was subsequently discovered by Burton (1969).

A *burst* of errors of length b is defined as a sequence of b-bit errors, the first and last of which are 1. The *burst error correction capability* of a code is defined as 1 less than the length of the shortest uncorrectable burst. It is relatively easy to show that a systematic (n, k) code, which has $n - k$ parity check bits, can correct bursts of length $b < \lfloor \frac{1}{2}(n - k) \rfloor$.

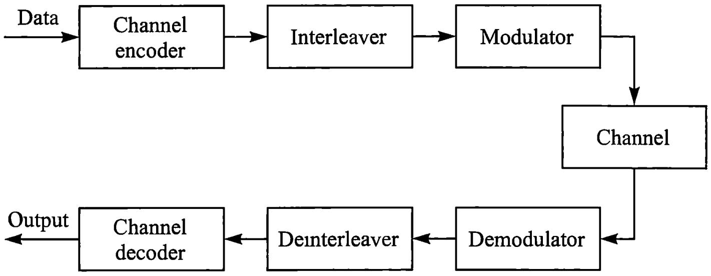

Data
Channel encoder
Interleaver
Modulator
Channel
Output
Channel decoder
Deinterleaver
Demodulator

FIGURE 7.12–1
Block diagram of system employing interleaving for burst error channel.

An effective method for dealing with burst error channels is to interleave the coded data in such a way that the bursty channel is transformed to a channel having independent errors. Thus, a code designed for independent channel errors (short bursts) is used.

A block diagram of a system that employs interleaving is shown in Figure 7.12–1. The encoded data are reordered by the interleaver and transmitted over the channel. At the receiver, after either hard or soft decision demodulation, the deinterleaver puts the data in proper sequence and passes them to the decoder. As a result of the interleaving/deinterleaving, error bursts are spread out in time so that errors within a codeword appear to be independent.

The interleaver can take one of two forms: a block structure or a convolutional structure. A block *interleaver* formats the encoded data in a rectangular array of m rows and n columns. Usually, each row of the array constitutes a codeword of length n. An *interleaver of degree m* consists of m rows (m codewords) as illustrated in Figure 7.12–2. The bits are read out columnwise and transmitted over the channel. At the receiver, the deinterleaver stores the data in the same rectangular array format, but they are read out rowwise, one codeword at a time. As a result of this reordering of the data during transmission, a burst of errors of length $l = mb$ is broken up into m bursts of length b. Thus, an (n, k) code that can handle burst errors of length $b < \lfloor \frac{1}{2}(n-k) \rfloor$ can be combined with an interleaver of degree m to create an interleaved (mn, mk) block code that can handle bursts of length mb. A *convolutional interleaver* can be used in place of a block interleaver in much the same way. Convolutional interleavers are better matched for

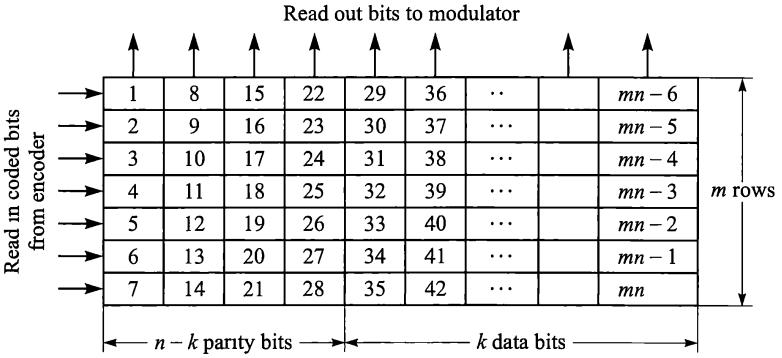

Read out bits to modulator
Read in coded bits from encoder
1 8 15 22 29 36 ·· mn − 6
2 9 16 23 30 37 ··· mn − 5
3 10 17 24 31 38 ··· mn − 4
4 11 18 25 32 39 ··· mn − 3
5 12 19 26 33 40 ··· mn − 2
6 13 20 27 34 41 ··· mn − 1
7 14 21 28 35 42 ··· mn
m rows
n − k parity bits
k data bits

FIGURE 7.12–2
A block interleaver for coded data.

use with the class of convolutional codes that is described in Chapter 8. Convolutional interleaver structures have been described by Ramsey (1970) and Forney (1971).

7.13
COMBINING CODES

The performance of a block code depends mainly on the number of errors it can correct, which is a function of the minimum distance of the code. For a given rate R_c, one can design codes with different block lengths. Codes with higher block length offer the possibility of higher minimum distances and thus higher error correction capability. This is clearly seen from the different bounds on the minimum distance derived in Section 7.7. The problem, however, is that the decoding complexity of a block code generally increases with the block length, and this dependence in general is an exponential dependence. Therefore improved performance through using block codes is achieved at the cost of increased decoding complexity.

One approach to design block codes with long block lengths and with manageable complexity is to begin with two or more simple codes with short block lengths and combine them in a certain way to obtain codes with longer block length that have better distance properties. Then some kind of suboptimal decoding can be applied to the combined code based on the decoding algorithms of the simple constituent codes.

7.13–1 Product Codes

A simple method of combining two or more codes is described in this section. The resulting codes are called *product codes*, first studied by Elias (1954). Let us assume we have two systematic linear block codes; code $\mathcal{C}_i$ is an (n_i, k_i) code with minimum distance $d_{\min i}$ for $i = 1, 2$. The product of these codes is an $(n_1 n_2, k_1 k_2)$ linear block code whose bits are arranged in a matrix form as shown in Figure 7.13–1.

The $k_1 k_2$ information bits are put in a rectangle with width k_1 and height k_2. The k_1 bits in each row of this matrix are encoded using the encoder for code $\mathcal{C}_1$, and the k_2 bits in each column are encoded using the encoder for code $\mathcal{C}_2$. The $(n_1 - k_1) \times (n_2 - k_2)$ bits

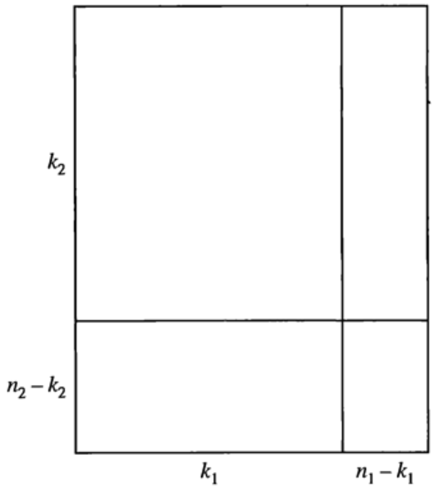


FIGURE 7.13–1
The structure of a product code.

in the lower right rectangle can be obtained either from encoding the bottom $n_2 - k_2$ rows using the encoding rule for $\mathcal{C}_1$ or from encoding the rightmost $n_1 - k_1$ columns using the encoding rule for $\mathcal{C}_2$. It is shown in Problem 7.63 that the results of these two approaches are the same.

The resulting code is an $(n_1 n_2,\ k_1 k_2)$ systematic linear block code. The rate of the product code is obviously the product of the rates of its component codes. Moreover, it can be shown that the minimum distance of the product code is the product of the minimum distances of the component codes, i.e., $d_{\min} = d_{\min 1} d_{\min 2}$ (see Problem 7.64), and hence the product code is capable of correcting

$$t = \left\lfloor \frac{d_{\min 1} d_{\min 2} - 1}{2} \right\rfloor \tag{7.13–1}$$

errors using a complex optimal decoding scheme.

We can design a simpler decoding scheme based on the decoding rules of the two constituent codes as follows. Let us assume

$$t_i = \left\lfloor \frac{d_{\min i} - 1}{2} \right\rfloor, \qquad i = 1, 2 \tag{7.13–2}$$

is the number of errors that code $\mathcal{C}_i$ can correct. Now let us assume in transmission of the $n_1 n_2$ binary digits of a codeword that fewer than $(t_1 + 1)(t_2 + 1)$ errors have occurred. Regardless of the location of errors, the number of rows of the product code shown in Figure 7.13–1 that have more than t_1 errors is less than or equal to t_2, because otherwise the total number of errors would be $(t_1 + 1)(t_2 + 1)$ or higher. Since each row having less than $t_1 + 1$ errors can be fully recovered using the decoding algorithm of $\mathcal{C}_1$, if we do rowwise decoding, we will have at most t_2 rows decoded erroneously. This means that after this stage of decoding the number of errors in each column cannot exceed t_2, all of which can be corrected using the decoding algorithm for $\mathcal{C}_2$ on columns. Therefore, using this simple two-stage decoding algorithm, we can correct up to

$$\begin{aligned} \tau &= (t_1 + 1)(t_2 + 1) - 1 \\ &= t_1 t_2 + t_1 + t_2 \end{aligned} \tag{7.13–3}$$

errors.

EXAMPLE 7.13–1. Consider a (255, 123) BCH code with $d_{\min 1} = 39$ and $t_1 = 19$ and a (15, 7) BCH code with $d_{\min 2} = 5$ and $t_2 = 2$ (see Example 7.10–3). The product of these codes has a minimum distance of $39 \times 5 = 195$ and can correct up to 97 errors if a complex decoding algorithm is employed to take advantage of the full error-correcting capability of the code. A two-stage decoding algorithm can, however, correct up to $(19 + 1)(2 + 1) - 1 = 59$ errors at noticeably lower complexity.

Another decoding algorithm, similar to how a crossword puzzle is solved, can also be used for decoding product codes. Using the row codes, we can come up with the best guess for the bit values; and then using the column codes, we can improve these guesses. This process can be repeated in an iterative fashion, improving the quality of the guess in each step. This process is known as *iterative decoding* and is very similar to the way a crossword puzzle is solved. To employ this decoding procedure, we need decoding schemes for the row and column codes that are capable of providing *guesses* about each individual bit. In other words, decoding schemes with soft outputs — usually, the likelihood values — are desirable. We will describe such decoding procedures in our discussion of turbo codes in Chapter 8.

7.13–2 Concatenated Codes

In concatenated coding two codes, one binary and one nonbinary are concatenated such that the codewords of the binary code are treated as symbols of the nonbinary code. The combination of the binary channel and the binary encoder and decoder appears as a nonbinary channel to the nonbinary encoder and decoder. The binary code that is directly connected to the binary channel is called the *inner code*, and the nonbinary code that operates on the combination of binary encoder/binary channel/binary decoder is called the *outer code*.

To be more specific, let us consider the concatenated coding scheme shown in Figure 7.13–2. The nonbinary (N, K) code forms the outer code, and the binary code forms the inner code. Codewords are formed by subdividing a block of kK information bits into K groups, called *symbols*, where each symbol consists of k bits. The K k-bit symbols are encoded into N k-bit symbols by the outer encoder, as is usually done with a nonbinary code. The inner encoder takes each k-bit symbol and encodes it into a binary block code of length n. Thus we obtain a concatenated block code having a block length of Nn bits and containing kK information bits. That is, we have created an equivalent (Nn, Kk) long binary code. The bits in each codeword are transmitted over the channel by means of PSK or, perhaps, by FSK.

We also indicate that the minimum distance of the concatenated code is $d_{\min} D_{\min}$, where $D_{\min}$ is the minimum distance of the outer code and $d_{\min}$ is the minimum distance of the inner code. Furthermore, the rate of the concatenated code is Kk/Nn, which is equal to the product of the two code rates.

A hard decision decoder for a concatenated code is conveniently separated into an inner decoder and an outer decoder. The inner decoder takes the hard decisions on each group of n bits, corresponding to a codeword of the inner code, and makes a decision on the k information bits based on maximum-likelihood (minimum-distance) decoding. These k bits represent one symbol of the outer code. When a block of N k-bit symbols is received from the inner decoder, the outer decoder makes a hard decision on the K k-bit symbols based on maximum-likelihood decoding.

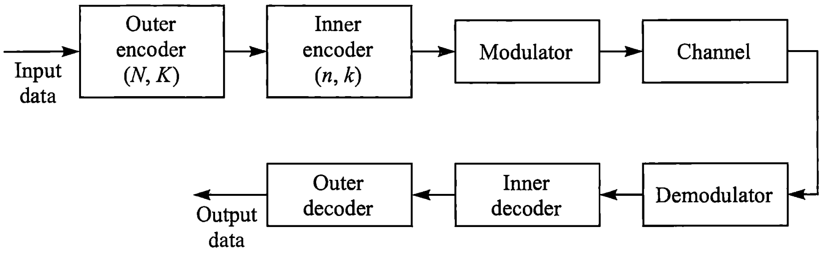


FIGURE 7.13–2
A concatenated coding scheme.

Soft decision decoding is also a possible alternative with a concatenated code. Usually, the soft decision decoding is performed on the inner code, if it is selected to have relatively few codewords, i.e., if 2^k is not too large. The outer code is usually decoded by means of hard decision decoding, especially if the block length is long and there are many codewords. On the other hand, there may be a significant gain in performance when soft decision decoding is used on both the outer and inner codes, to justify the additional decoding complexity. This is the case in digital communications over fading channels, as we shall demonstrate in Chapter 14.

EXAMPLE 7.13–2. Suppose that the (7, 4) Hamming code is used as the inner code in a concatenated code in which the outer code is a Reed-Solomon code. Since $k = 4$, we select the length of the Reed-Solomon code to be $N = 2^4 - 1 = 15$. The number of information symbols K per outer codeword may be selected over the range $1 \leq K \leq 14$ in order to achieve a desired code rate.

Concatenated codes with Reed-Solomon codes as the outer code and binary convolutional codes as the inner code have been widely used in the design of deep space communication systems. More details on concatenated codes can be found in the book by Forney (1966a).

Serial and Parallel Concatenation with Interleavers

An interleaver may be used in conjunction with a concatenated code to construct a code with extremely long codewords. In a *serially concatenated block code* (SCBC), the interleaver is inserted between the two encoders as shown in Figure 7.13–3. Both codes are linear systematic binary codes. The outer code is a (p, k) code, and the inner code is an (n, p) code. The block interleaver length is selected as $N = mp$, where m is a usually large positive integer that determines the overall block length. The encoding and interleaving are performed as follows: mk information bits are encoded by the outer encoder to produce mp coded bits. These $N = mp$ coded bits are read out of the interleaver in different order according to the permutation algorithm of the interleaver. The mp bits at the output of the interleaver are fed to the inner encoder in blocks of length p. Therefore, a block of mk information bits is encoded by the SCBC into a block of mn bits. The resulting code rate is $R_c^s = k/n$, which is the product of the code rates of the inner and outer encoders. However, the block length of the SCBC is nm bits, which can be significantly larger than the block length of the conventional serial concatenation of the block codes without the use of the interleaver.

The block interleaver is usually implemented as a pseudorandom interleaver, i.e., an interleaver that pseudorandomly permutes the block of N bits. For purposes of analyzing the performance of SCBC, such an interleaver may be modeled as a *uniform*

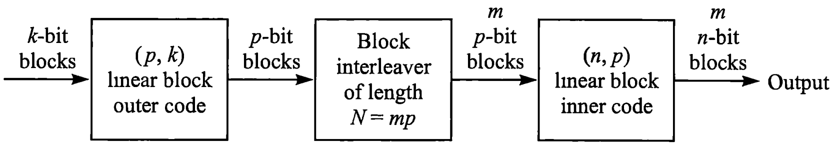

k-bit blocks
(p, k) linear block outer code
p-bit blocks
Block interleaver of length $N = mp$
m p-bit blocks
(n, p) linear block inner code
m n-bit blocks
Output

FIGURE 7.13–3
Serial concatenated block code with interleaver.

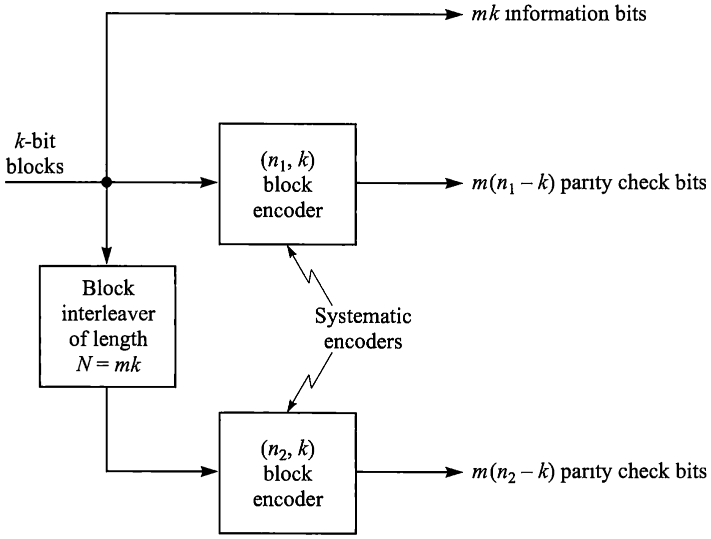


FIGURE 7.13–4
Parallel concatenated block code (PCBC) with interleaver.

interleaver, which is defined as a device that maps a given input word of weight w into all distinct $\binom{N}{w}$ permutations with equal probability. This operation is similar to Shannon's random coding argument, where here the average performance is measured over all possible interleavers of length N.

By use of interleaving, *parallel concatenated block codes* (PCBCs) can be constructed in a similar manner. Figure 7.13–4 illustrates the basic configuration of such an encoder based on two constituent binary codes. The constituent codes may be identical or different. The two encoders are systematic, binary linear encoders, denoted as (n_1, k) and (n_2, k). The pseudorandom block interleaver has length $N = k$, and thus the overall PCBC has block length $n_1 + n_2 - k$ and rate $k/(n_1 + n_2 - k)$, since the information bits are transmitted only once. More generally, we may encode mk bits $(m > 1)$ and thus use an interleaver of length $N = mk$. The design of interleavers for parallel concatenated codes is considered in a paper by Daneshgaran and Mondin (1999).

The use of an interleaver in the construction of SCBC and PCBC results in codewords that are both large in block length and relatively sparse. Decoding of these types of codes is generally performed iteratively, using soft-in/soft-out (SISO) maximum a posteriori probability (MAP) algorithms. An iterative MAP decoding algorithm for serially concatenated codes is described in the paper by Benedetto et al. (1998). Iterative MAP decoding algorithms for parallel concatenated codes have been described in a number of papers, including Berrou et al. (1993), Benedetto and Montorsi (1996), Hagenauer et al. (1996) and in the book by Heegard and Wicker (1999). The combination of code concatenation with interleaving and iterative MAP decoding results in performance very close to the Shannon limit at moderate error rates, such as 10^{-4} to 10^{-5} (low SNR region). More details on this type of concatenation will be given in Chapter 8.

7.14
BIBLIOGRAPHICAL NOTES AND REFERENCES

The pioneering work on coding and coded waveforms for digital communications was done by Shannon (1948), Hamming (1950), and Golay (1949). These works were rapidly followed with papers on code performance by Gilbert (1952), new codes by Muller (1954) and Reed (1954), and coding techniques for noisy channels by Elias (1954, 1955) and Slepian (1956). During the period 1960–1970, there were a number of significant contributions in the development of coding theory and decoding algorithms. In particular, we cite the papers by Reed and Solomon (1960) on Reed-Solomon codes, the papers by Hocquenghem (1959) and Bose and Ray-Chaudhuri (1960) on BCH codes, and the Ph.D. dissertation of Forney (1966) on concatenated codes. These works were followed by the papers of Goppa (1970, 1971) on the construction of a new class of linear cyclic codes, now called Goppa codes [see also Berlekamp (1973)], and the paper of Justesen (1972) on a constructive technique for asymptotically good codes. During this period, work on decoding algorithms was primarily focused on BCH codes. The first decoding algorithm for binary BCH codes was developed by Peterson (1960). A number of refinements and generalizations by Chien (1964), Forney (1965), Massey (1965), and Berlekamp (1968) led to the development of the Berlekamp-Massey algorithm described in detail in Lin and Costello (2004) and Wicker (1995). A treatment of Reed-Solomon codes is given in the book by Wicker and Bhargava (1994).

In addition to the references given above on coding, decoding, and coded signal design, we should mention the collection of papers published by the IEEE Press entitled *Key Papers in the Development of Coding Theory*, edited by Berlekamp (1974). This book contains important papers that were published in the first 25 years of the development of coding theory. We should also cite the Special Issue on Error-Correcting Codes, *IEEE Transactions on Communications* (October 1971). Finally, the survey papers by Calderbank (1998), Costello et al. (1998), and Forney and Ungerboeck (1998) highlight the major developments in coding and decoding over the past 50 years and include a large number of references. Standard textbooks on this subject include those by Lin and Costello (2004), MacWilliams and Sloane (1977), Blahut (2003), Wicker (1995), and Berlekamp (1968).

PROBLEMS

7.1 From the definition of a Galois field GF(q) we know that $\{F - \{0\}, \cdot, 1\}$ is an Abelian group with $q - 1$ elements.

1. Let $a \in \{F - \{0\}, \cdot, 1\}$ and define $a^i = \underbrace{a \cdot a \cdot a \cdots a}_{i \text{ times}}$. Show that for some positive j we have $a^j = 1$ and $a^i \neq 1$ for all $0 < i < j$, where j is called the *order* of a.
2. Show that if $0 < i < i' \leq j$, then a^i and $a^{i'}$ are distinct elements of $\{F - \{0\}, \cdot, 1\}$.
3. Show that $\mathcal{G}_a = \{a, a^2, a^3, \ldots, a^j\}$ is an Abelian group under multiplication; $\mathcal{G}_a$ is called the *cyclic subgroup* of element a.

4. Let us assume that a $b \in \{F - \{0\}, \cdot, 1\}$ exists such that $b \notin \mathcal{G}_a$. Show that $\mathcal{G}_{ba} = \{b \cdot a, b \cdot a^2, \ldots, b \cdot a^j\}$ is an Abelian group and $\mathcal{G}_a \cap \mathcal{G}_{ba} = \emptyset$. Therefore, if such a b exists, the number of elements in $\{F - \{0\}, \cdot, 1\}$ is at least $2j$, and $\mathcal{G}_{ba}$ is called a *coset* of $\mathcal{G}_a$.
5. Use the argument of part 4 to prove that the nonzero elements of GF(q) can be written as the union of disjoint cosets, and hence the order of any element of GF(q) divides $q - 1$.
6. Conclude that for any nonzero $\beta \in$ GF(q) we have $\beta^{q-1} = 1$.

7.2 Use the result of Problem 7.1 to prove that the q elements of GF(q) are the roots of equation

$$X^q - X = 0$$

7.3 Construct the addition and multiplication tables of GF(5).

7.4 List all prime polynomials of degrees 2 and 3 over GF(3). Using a prime polynomial of degree 2, generate the multiplication table of GF(9).

7.5 List all primitive elements in GF(8). How many primitive elements are in GF(32)?

7.6 Let $\alpha \in$ GF(2^4) be a primitive element. Show that $\{0, 1, \alpha^5, \alpha^{10}\}$ is a field. From this conclude that GF(4) is a *subfield* of GF(16).

7.7 Show that GF(4) is *not* a subfield of GF(32).

7.8 Using Table 7.1–5, generate GF(32) and express its elements in polynomials, power, and vector form. Find the minimal polynomials of $\beta = \alpha^3$ and $\gamma = \alpha^3 + \alpha$, where α is a primitive element.

7.9 Let $\beta \in$ GF(p^m) be a nonzero element. Show that

$$\sum_{i=1}^{p} \beta = 0$$

and

$$\sum_{i=1}^{m} \beta \neq 0$$

for all $0 < m < p$.

7.10 Let $\alpha, \beta \in$ GF(p^m). Show that

$$(\alpha + \beta)^p = \alpha^p + \beta^p$$

7.11 Show that any binary linear block code of length n has exactly 2^k codewords for some integer $k \leq n$.

7.12 Prove that the Hamming distance between two sequences of length n, denoted by $d_H(\boldsymbol{x}, \boldsymbol{y})$, satisfies the following properties:

1. $d_H(\boldsymbol{x}, \boldsymbol{y}) = 0$ if and only if $\boldsymbol{x} = \boldsymbol{y}$

2. $d_H(\boldsymbol{x}, \boldsymbol{y}) = d_H(\boldsymbol{y}, \boldsymbol{x})$
3. $d_H(\boldsymbol{x}, \boldsymbol{z}) \leq d_H(\boldsymbol{x}, \boldsymbol{y}) + d_H(\boldsymbol{y}, \boldsymbol{z})$

These properties show that d_H is a metric.

7.13 The generator matrix for a linear binary code is

$$\boldsymbol{G} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

a. Express $\boldsymbol{G}$ in systematic $[\boldsymbol{I}\,|\,\boldsymbol{P}]$ form.
b. Determine the parity check matrix $\boldsymbol{H}$ for the code.
c. Construct the table of syndromes for the code.
d. Determine the minimum distance of the code.
e. Demonstrate that the codeword $\boldsymbol{c}$ corresponding to the information sequence 101 satisfies $\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0}$.

7.14 A code is self-dual if $\mathcal{C} = \mathcal{C}^\perp$. Show that in a self-dual code the block length is always even and the rate is $\frac{1}{2}$.

7.15 Consider a linear block code with codewords $\{0000, 1010, 0101, 1111\}$. Find the dual of this code and show that this code is self-dual.

7.16 List the codewords generated by the matrices given in Equations 7.9–13 and 7.9–15, and thus demonstrate that these matrices generate the same set of codewords.

7.17 Determine the weight distribution of the (7, 4) Hamming code, and check your result with the list of codewords given in Table 7.9–2.

7.18 Show that for binary orthogonal signaling, for instance, orthogonal BFSK, we have $\Delta = e^{-\mathcal{E}_c/2N_0}$, where Δ is defined by Equation 7.2–36.

7.19 Find the generator and the parity check matrices of a second-order ($r = 2$) Reed-Muller code with block length $n = 16$. Show that this code is the dual of a first-order Reed-Muller code with $n = 16$.

7.20 Show that repetition codes whose block length is a power of 2 are Reed-Muller codes of order $r = 0$.

7.21 When an (n, k) Hadamard code is mapped into waveforms by means of binary PSK, the corresponding $M = 2^k$ waveforms are orthogonal. Determine the bandwidth expansion factor for the M orthogonal waveforms, and compare this with the bandwidth requirements of orthogonal FSK detected coherently.

7.22 Show that the signaling waveforms generated from a maximum-length shift register code by mapping each bit in a codeword into a binary PSK signal are equicorrelated with correlation coefficient $\rho_r = -1/(M-1)$, i.e., the M waveforms form a simplex set.

7.23 Using the generator matrix of a $(2^m - 1, m)$ maximum-length code as defined in Section 7.3–3, do the following.

a. Show that maximum-length codes are constant-weight codes; i.e., all nonzero codewords of a $(2^m - 1, m)$ maximum-length code have weight 2^{m-1}.

b. Show that the weight distribution function of a maximum-length code is given by Equation 7.3–4.

c. Use the MacWilliams identity to determine the weight distribution function of a $(2^m - 1, 2^m - 1 - m)$ Hamming code as the dual to a maximum-length code.

7.24 Compute the error probability obtained with a (7, 4) Hamming code on an AWGN channel, for both hard decision and soft decision decoding. Use Equations 7.4–18, 7.4–19, 7.5–6, and 7.5–18.

7.25 Show that when a binary sequence $\boldsymbol{x}$ of length n is transmitted over a BSC with crossover probability p, the probability of receiving $\boldsymbol{y}$, which is at Hamming distance d from $\boldsymbol{x}$, is given by

$$P(\boldsymbol{y}|\boldsymbol{x}) = (1-p)^n \left(\frac{p}{1-p}\right)^d$$

From this conclude that if $p < \frac{1}{2}$, $P(\boldsymbol{y}|\boldsymbol{x})$ is a decreasing function of d and hence ML decoding is equivalent to minimum-Hamming-distance decoding. What happens if $p > \frac{1}{2}$?

7.26 Using a symbolic computation program (e.g., Mathematica or Maple), find the weight enumeration polynomial for a (15, 11) Hamming code. Plot the probability of decoding error (when this code is used for error correction) and undetected error (when the code used for error detection) as a function of the channel error probability p in the range $10^{-6} \le p \le 10^{-1}$.

7.27 By using a computer find the number of codewords of weight 34 in a (63, 57) Hamming code.

7.28 Prove that if the sum of two error patterns $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ is a valid codeword $\boldsymbol{c}_j$, then each error pattern has the same syndrome.

7.29 Prove that any two n-tuples in the same row of a standard array add to produce a valid codeword.

7.30 Prove that

1. Elements of the standard array of a linear block code are distinct.
2. Two elements belonging to two distinct cosets of a standard array have distinct syndromes.

7.31 A $(k+1, k)$ block code is generated by adding 1 extra bit to each information sequence of length k such that the overall parity of the code (i.e., the number of 1s in each codeword) is an odd number. Two students, A and B, make the following arguments on *error detection* capability of this code.

1. Student A: Since the the weight of each codeword is odd, any single error changes the weight to an even number. Hence, this code is capable of detecting any single error.

2. Student B: The all-zero information sequence $\underbrace{00\cdots0}_{k}$ will be encoded by adding one extra 1 to generate the codeword $\underbrace{00\cdots0}_{k}1$. This means that there is at least one codeword of weight 1 in this code. Therefore, $d_{\min} = 1$, and since any code can detect at most $d_{\min} - 1$ errors, and for this code $d_{\min} - 1 = 0$, this code cannot detect any errors.

Which argument do you agree with and why? Give your explanation in one short paragraph.

7.32 The parity check matrix of a linear block code is given below:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

1. Determine the generator matrix for this code in the systematic form.
2. How many codewords are in this code? What is the $d_{\min}$ for this code?
3. What is the coding gain for this code (soft decision decoding and BPSK modulation over an AWGN channel are assumed)?
4. Using hard decision decoding, how many errors can this code correct?
5. Show that any two codewords of this code are orthogonal, and in particular any codeword is orthogonal to itself.

7.33 A code $\mathcal{C}$ consists of all binary sequences of length 6 and weight 3.

1. Is this code a linear block code? Why?
2. What is the rate of this code? What is the minimum distance of this code? What is the minimum weight for this code?
3. If the code is used for error detection, how many errors can it detect?
4. If the code is used on a binary symmetric channel with crossover probability of p, what is the probability that an undetectable error occurs?
5. Find the smallest linear block code $\mathcal{C}_1$ such that $\mathcal{C} \subseteq \mathcal{C}_1$ (by the smallest code we mean the code with the fewest codewords).

7.34 A systematic (6, 3) code has the generator matrix

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Construct the standard array and determine the correctable error patterns and their corresponding syndromes.

7.35 Construct the standard array for the (7, 3) code with generator matrix

$$\boldsymbol{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

and determine the correctable patterns and their corresponding syndromes.

7.36 A (6, 3) *systematic* linear block code encodes the information sequence $\boldsymbol{x} = (x_1, x_2, x_3)$ into codeword $\boldsymbol{c} = (c_1, c_2, c_3, c_4, c_5, c_6)$, such that c_4 is a parity check on c_1 and c_2, to make the overall parity even (i.e., $c_{\mathrm{i}} \oplus c_2 \oplus c_4 = 0$). Similarly c_5 is a parity check on c_2 and c_3, and c_6 is a parity check on c_1 and c_3.

1. Determine the generator matrix of this code.
2. Find the parity check matrix for this code.
3. Using the parity check matrix, determine the minimum distance of this code.
4. How many errors is this code capable of correcting?
5. If the received sequence (using hard decision decoding) is $\boldsymbol{y} = 100000$, what is the transmitted sequence using a maximum-likelihood decoder? (Assume that the crossover probability of the channel is less than $\frac{1}{2}$.)

7.37 $\mathcal{C}$ is a (6, 3) linear block code whose generator matrix is given by

$$\boldsymbol{G} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

1. What rate, minimum distance, and the coding gain can $\mathcal{C}$ provide in *soft decision* decoding when BPSK is used over an AWGN channel?
2. Can you suggest another (6, 3) LBC that can provide a better coding gain? If the answer is yes, what is its generator matrix and the resulting coding gain? If the answer is no, why?
3. Suggest a parity check matrix $\boldsymbol{H}$ for $\mathcal{C}$.

7.38 Prove that if $\mathcal{C}$ is MDS, its dual $\mathcal{C}^{\perp}$ is also MDS.

7.39 Let n and t be positive integers such that $n > 2t$; hence $\frac{t}{n} < \frac{1}{2}$.

1. Show that for any $\lambda > 0$ we have

$$2^{\lambda(n-t)} \sum_{i=0}^{t} \binom{n}{i} \leq \sum_{i=n-t}^{n} 2^{\lambda i} \binom{n}{i} \leq (1 + 2^{\lambda})^n$$

2. Assuming $p = t/n$ in part 1, show that

$$\sum_{i=0}^{n} \binom{n}{i} \leq (2^{-\lambda(1-p)} + 2^{\lambda p})^n$$

3. By choosing $\lambda = \log_2 \frac{1-p}{p}$ show that

$$\sum_{i=0}^{n} \binom{n}{i} \leq 2^{n H_b(p)}$$

4. Using Stirling's approximation that states that

$$n! = \sqrt{2\pi n} \left(\frac{n}{e}\right)^n e^{\lambda_n}$$

where $\frac{1}{12n+1} < \lambda_n < \frac{1}{12n}$, show that for large n and t such that $\frac{t}{n} < \frac{1}{2}$ we have

$$\sum_{i=0}^{t} \binom{n}{i} \approx 2^{nH_b\left(\frac{t}{n}\right)}$$

7.40 Let $\mathcal{C}$ denote an (n, k) linear block code with minimum distance $d_{\min}$.

a. Let $\boldsymbol{C}$ denote a $2^k \times n$ matrix whose rows are all the codewords of $\mathcal{C}$. Show that all *columns* of $\boldsymbol{C}$ have equal weight and this weight is 2^{k-1}.

b. Conclude that the total weight of the codewords of $\mathcal{C}$ is given by

$$d_{\text{total}} = \sum_{m=1}^{2^k} w\left(\boldsymbol{c}_m\right) = n2^{k-1}$$

c. From part (b) conclude that the Plotkin bound

$$d_{\min} \le \frac{n\,2^{k-1}}{2^k - 1}$$

7.41 Construct an extended $(8, 4)$ code from the $(7, 4)$ Hamming code by specifying the generator matrix and the parity check matrix.

7.42 The polynomial

$$g(X) = X^4 + X + 1$$

is the generator for the $(15, 11)$ Hamming binary code.

a. Determine a generator matrix $\boldsymbol{G}$ for this code in systematic form.

b. Determine the generator polynomial for the dual code.

7.43 For the $(7, 4)$ cyclic Hamming code with generator polynomial $g(X) = X^3 + X^2 + 1$, construct an $(8, 4)$ extended Hamming code and list all the codewords. What is $d_{\min}$ for the extended code?

7.44 An $(8, 4)$ linear block code is constructed by shortening a $(15, 11)$ Hamming code generated by the generator polynomial $g(X) = X^4 + X + 1$.

a. Construct the codewords of the $(8, 4)$ code and list them.

b. What is the minimum distance of the $(8, 4)$ code?

7.45 The polynomial $X^{15} + 1$ when factored yields

$$X^{15} + 1 = (X^4 + X^3 + 1)(X^4 + X^3 + X^2 + X + 1)(X^4 + X + 1)(X^2 + X + 1)(X + 1)$$

a. Construct a systematic $(15, 5)$ code using the generator polynomial

$$g(X) = (X^4 + X^3 + X^2 + X + 1)(X^4 + X + 1)(X^2 + X + 1)$$

b. What is the minimum distance of the code?

c. How many random errors per codeword can be corrected?

d. How many errors can be detected by this code?

e. List the codewords of a $(15, 2)$ code constructed from the generator polynomial

$$g(X) = \frac{X^{15}+1}{X^2+X+1}$$

and determine the minimum distance.

7.46 Construct the parity check matrices $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ corresponding to the generator matrices $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ given by Equations 7.9–12 and 7.9–13, respectively.

7.47 Determine the correctable error patterns (of least weight) and their syndromes for the systematic $(7, 4)$ cyclic Hamming code.

7.48 Let $g(X) = X^8 + X^6 + X^4 + X^2 + 1$ be a polynomial over the binary field.

- *a.* Find the lowest-rate cyclic code with generator polynomial $g(X)$. What is the rate of this code?
- *b.* Find the minimum distance of the code found in (a).
- *c.* What is the coding gain for the code found in (a)?

7.49 The polynomial $g(X) = X + 1$ over the binary field is considered.

- *a.* Show that this polynomial can generate a cyclic code for any choice of n. Find the corresponding k.
- *b.* Find the systematic form of $\boldsymbol{G}$ and $\boldsymbol{H}$ for the code generated by $g(X)$.
- *c.* Can you say what type of code this generator polynomial generates?

7.50 Design a $(6, 2)$ cyclic code by choosing the shortest possible generator polynomial.

- *a.* Determine the generator matrix $\boldsymbol{G}$ (in the systematic form) for this code, and find all possible codewords.
- *b.* How many errors can be corrected by this code?

7.51 Let $\mathcal{C}_1$ and $\mathcal{C}_2$ denote two cyclic codes with the same block length n, with generator polynomials $g_1(X)$ and $g_2(X)$, and with minimum distances d_1 and d_2, respectively. Define $\mathcal{C}_{\max} = \mathcal{C}_1 \cup \mathcal{C}_2$ and $\mathcal{C}_{\min} = \mathcal{C}_1 \cap \mathcal{C}_2$.

1. Is $\mathcal{C}_{\max}$ a cyclic code? Why? If yes, what is its generator polynomial and its minimum distance?
2. Is $\mathcal{C}_{\min}$ a cyclic code? Why? If yes, find its generator polynomial. What can you say about its minimum distance?

7.52 We know that cyclic codes for all possible values of (n, k) do not exist.

1. Give an example of an (n, k) pair for which no cyclic code exists ($k < n$).
2. How many $(10, 2)$ cyclic codes do exist? Determine the generator polynomial of one such code.
3. Determine the minimum distance of the code in part 2.
4. How many errors can the code in part 2 correct?
5. If this code is employed for transmission over a channel which uses binary antipodal signaling with hard decision decoding and the SNR per bit of the channel is $\gamma_b = 3$ dB, determine an upper bound on the error probability of the system.

7.53 What are the possible rates for cyclic codes with block length 23? List all possible generator polynomials and specify the generator polynomial of the $(23, 12)$ Golay code.

7.54 Let $s(X)$ denote the syndrome corresponding to error sequence $e(X)$ in an (n, k) cyclic code with generator polynomial $g(X)$. Show that the syndrome corresponding to $e^{(1)}(X)$, the right cyclic shift of $e(X)$, is $s^{(1)}(X)$, defined by

$$s^{(1)}(X) = Xs(X) \mod g(X)$$

7.55 Is the following statement true or false? If it is true, prove it; and if it is false, give a counterexample: The minimum weight of a cyclic code is equal to the number of nonzero coefficients of its generator polynomial.

7.56 Determine the generator polynomial and the rate of a double-error-correcting BCH code with block length $n = 31$.

7.57 In the BCH code designed in Problem 7.56 the received sequence is

$$\boldsymbol{r} = 0000000000000000000011001001001$$

Using the Berlekamp-Massey algorithm, detect the error locations.

7.58 Solve Problem 7.57 when the received sequence is

$$\boldsymbol{r} = 1110000000000000000011101101001$$

7.59 Beginning with a $(15, 7)$ BCH code, construct a shortened $(12, 4)$ code. Give the generator matrix for the shortened code.

7.60 Determine the generator polynomial and the rate of a double-error-correcting Reed-Solomon code with block length $n = 7$.

7.61 Determine the generator polynomial and the rate of a triple-error-correcting Reed-Solomon code with block length $n = 63$. How many codewords does this code have?

7.62 What is the weight distribution function of the Reed-Solomon code designed in Problem 7.60?

7.63 Prove that in the product code shown in Figure 7.13–1 the $(n_1 - k_1) \times (n_2 - k_2)$ bits in the lower right corner can be obtained as either the parity checks on the rows or parity checks on the columns.

7.64 Prove that the minimum distance of a product code is the product of the minimum distances of the two constituent codes.

8

Trellis and Graph Based Codes

Linear block codes were studied in detail in Chapter 7. These codes are mainly used with hard decision decoding that employs the built-in algebraic structure of the code based on the properties of finite fields. Hard decision decoding of these codes results in a binary symmetric channel model consisting of the binary modulator, the waveform channel, and the optimum binary detector. The decoder for these codes tries to find the codeword at the minimum Hamming distance from the output of the BSC. The goal in designing good linear block codes is to find the code with highest minimum distance for a given n and k.

In this chapter we introduce another class of codes whose structure is more conveniently described in terms of trellises or graphs. We will see that for this family of codes, soft decision decoding is possible, and in some cases performance very close to channel capacity is achievable.

8.1
THE STRUCTURE OF CONVOLUTIONAL CODES

A convolutional code is generated by passing the information sequence to be transmitted through a linear finite-state shift register. In general, the shift register consists of K (k-bit) stages and n linear algebraic function generators, as shown in Figure 8.1–1. The input data to the encoder, which is assumed to be binary, is shifted into and along the shift register k bits at a time. The number of output bits for each k-bit input sequence is n bits. Consequently, the code rate is defined as $R_c = k/n$, consistent with the definition of the code rate for a block code. The parameter K is called the *constraint length* of the convolution code.†

---

†In many cases, the constraint length of the code is given in bits rather than k-bit bytes. Hence, the shift register may be called an *L-stage shift register*, where $L = Kk$. Furthermore, L may not be a multiple of k, in general.

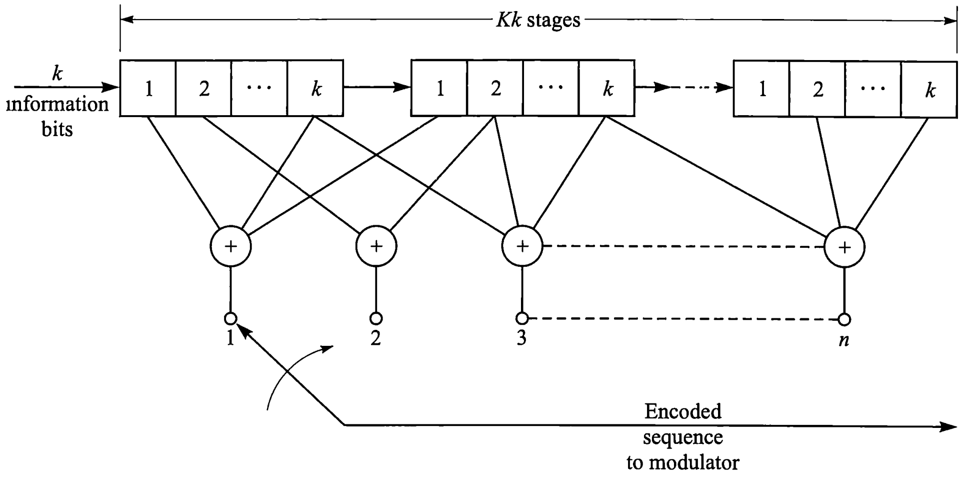


FIGURE 8.1–1
Convolutional encoder.

One method for describing a convolutional code is to give its generator matrix, just as we did for block codes. In general, the generator matrix for a convolutional code is semi-infinite since the input sequence is semi-infinite in length. As an alternative to specifying the generator matrix, we shall use a functionally equivalent representation in which we specify a set of n vectors, one vector for each of the n modulo-2 adders. Each vector has Kk dimensions and contains the connections of the encoder to that modulo-2 adder. A 1 in the ith position of the vector indicates that the corresponding stage in the shift register is connected to the modulo-2 adder, and a 0 in a given position indicates that no connection exists between that stage and the modulo-2 adder.

To be specific, let us consider the binary convolutional encoder with constraint length $K = 3, k = 1$, and $n = 3$, which is shown in Figure 8.1–2. Initially, the shift register is assumed to be in the all-zeros state. Suppose the first input bit is a 1. Then the output sequence of 3 bits is 111. Suppose the second bit is a 0. The output sequence will then be 001. If the third bit is a 1, the output will be 100, and so on. Now, suppose we number the outputs of the function generators that generate each 3-bit output sequence as 1, 2, and 3, from top to bottom, and similarly number each corresponding function generator. Then, since only the first stage is connected to the first function generator (no modulo-2 adder is needed), the generator is

$$g_1 = [100]$$

The second function generator is connected to stages 1 and 3. Hence

$$g_2 = [101]$$

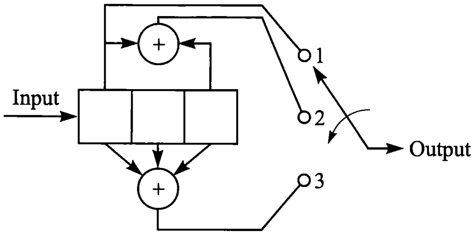


FIGURE 8.1–2
$K = 3, k = 1, n = 3$ convolutional encoder.

Finally,

$$g_3 = [111]$$

The generators for this code are more conveniently given in octal form as (4, 5, 7). We conclude that when $k = 1$, we require n generators, each of dimension K to specify the encoder.

It is clear that $\boldsymbol{g}_1$, $\boldsymbol{g}_2$, and $\boldsymbol{g}_3$ are the impulse responses from the encoder input to the three outputs. Then if the input to the encoder is the information sequence $\boldsymbol{u}$, the three outputs are given by

$$\begin{aligned} \boldsymbol{c}^{(1)} &= \boldsymbol{u} \star \boldsymbol{g}_1 \\ \boldsymbol{c}^{(2)} &= \boldsymbol{u} \star \boldsymbol{g}_2 \\ \boldsymbol{c}^{(3)} &= \boldsymbol{u} \star \boldsymbol{g}_3 \end{aligned} \tag{8.1–1}$$

where $\star$ denotes the convolution operation. The corresponding code sequence $\boldsymbol{c}$ is the result of interleaving $\boldsymbol{c}^{(1)}$, $\boldsymbol{c}^{(2)}$, and $\boldsymbol{c}^{(3)}$ as

$$\boldsymbol{c} = \left(c_1^{(1)}, c_1^{(2)}, c_1^{(3)}, c_2^{(1)}, c_2^{(2)}, c_2^{(3)}, \ldots\right) \tag{8.1–2}$$

The convolutional operation is equivalent to multiplication in the transform domain. We define the D transform† of $\boldsymbol{u}$ as

$$u(D) = \sum_{i=0}^{\infty} u_i D^i \tag{8.1–3}$$

and the transfer function for the three impulse responses $\boldsymbol{g}_1$, $\boldsymbol{g}_2$, and $\boldsymbol{g}_3$ as

$$\begin{aligned} g_1(D) &= 1 \\ g_2(D) &= 1 + D^2 \\ g_3(D) &= 1 + D + D^2 \end{aligned} \tag{8.1–4}$$

The output transforms are then given by

$$\begin{aligned} c^{(1)}(D) &= u(D)g_1(D) \\ c^{(2)}(D) &= u(D)g_2(D) \\ c^{(3)}(D) &= u(D)g_3(D) \end{aligned} \tag{8.1–5}$$

and the transform of the encoder output $\boldsymbol{c}$ is given by

$$c(D) = c^{(1)}(D^3) + Dc^{(2)}(D^3) + D^2c^{(3)}(D^3) \tag{8.1–6}$$

EXAMPLE 8.1–1. Let the sequence $\boldsymbol{u} = (100111)$ be the input sequence to the convolutional encoder shown in Figure 8.1–2. We have

$$u(D) = 1 + D^3 + D^4 + D^5$$

---

†Using the D transform is common in coding literature where D denotes the unit delay introduced by one memory element in the shift register. By substituting $D = z^{-1}$, the D transform becomes the familiar z transform.

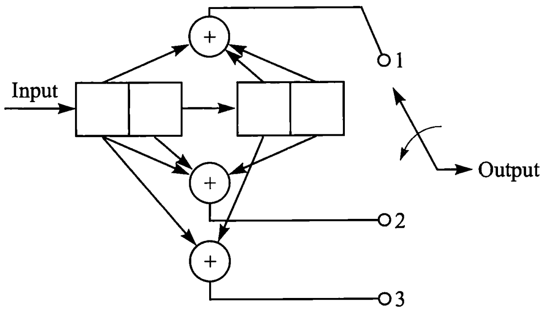


FIGURE 8.1–3
$K = 2, k = 2, n = 3$ convolutional encoder.

and

$$c^{(1)}(D) = (1 + D^3 + D^4 + D^5)(1) = 1 + D^3 + D^4 + D^5$$
$$c^{(2)}(D) = (1 + D^3 + D^4 + D^5)(1 + D^2) = 1 + D^2 + D^3 + D^4 + D^6 + D^7$$
$$c^{(3)}(D) = (1 + D^3 + D^4 + D^5)(1 + D + D^2) = 1 + D + D^2 + D^3 + D^5 + D^7$$

and

$$\begin{aligned} c(D) &= c^{(1)}(D^3) + Dc^{(2)}(D^3) + D^2c^{(3)}(D^3) \\ &= 1 + D + D^2 + D^5 + D^7 + D^8 + D^9 + D^{10} + D^{11} + D^{12} + D^{13} + D^{15} \\ &\quad + D^{17} + D^{19} + D^{22} + D^{23} \end{aligned}$$

corresponding to the code sequence

$$\boldsymbol{c} = (111001011111110101010011)$$

For a rate k/n binary convolutional code with $k > 1$ and constraint length K, the n generators are Kk-dimensional vectors, as stated above. The following example illustrates the case in which $k = 2$ and $n = 3$.

EXAMPLE 8.1–2. Consider the rate 2/3 convolutional encoder illustrated in Figure 8.1–3. In this encoder, 2 bits at a time are shifted into it, and 3 output bits are generated. The generators are

$$\boldsymbol{g}_1 = [1011], \qquad \boldsymbol{g}_2 = [1101], \qquad \boldsymbol{g}_3 = [1010]$$

In octal form, these generators are (13, 15, 12).

The code shown in Figure 8.1–3 can be also realized by the diagram shown in Figure 8.1–4. In this realization, instead a single shift register of length 4, two shift registers each of length 2 are employed. The information sequence $\boldsymbol{u}$ is split into two substreams $\boldsymbol{u}^{(1)}$ and $\boldsymbol{u}^{(2)}$ using a serial-to-parallel converter. Each of the two substreams

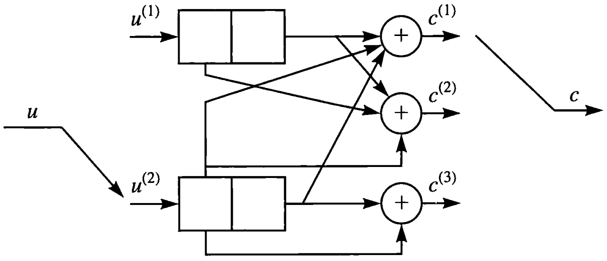


FIGURE 8.1–4
Double shift register implementation of the convolutional encoder shown in Figure 8.1–3.

is the input to one of the two shift registers. At the output, the three generated sequences $\boldsymbol{c}^{(1)}$, $\boldsymbol{c}^{(2)}$, and $\boldsymbol{c}^{(3)}$ are interleaved to generate the code sequence $\boldsymbol{c}$. In general, instead of one shift register with length $L = Kk$, we can use a parallel implementation of k shift registers each of length K.

In the implementation shown in Figure 8.1–4, the encoder has two input sequences $\boldsymbol{u}^{(1)}$ and $\boldsymbol{u}^{(2)}$ and three output sequences $\boldsymbol{c}^{(1)}$, $\boldsymbol{c}^{(2)}$, and $\boldsymbol{c}^{(3)}$. The encoder thus can be described in terms of six impulse responses, and hence six transfer functions which are the D transforms of the impulse responses. If we denote by $\boldsymbol{g}_i^{(j)}$ the impulse response from input stream $\boldsymbol{u}^{(i)}$ to the output stream $\boldsymbol{c}^{(j)}$, in the encoder depicted in Figure 8.1–4 we have

$$\begin{aligned} \boldsymbol{g}_1^{(1)} &= [0\ 1] & \boldsymbol{g}_2^{(1)} &= [1\ 1] \\ \boldsymbol{g}_1^{(2)} &= [1\ 1] & \boldsymbol{g}_2^{(2)} &= [1\ 0] \\ \boldsymbol{g}_1^{(3)} &= [0\ 0] & \boldsymbol{g}_2^{(3)} &= [1\ 1] \end{aligned} \tag{8.1–7}$$

and the transfer functions are

$$\begin{aligned} g_1^{(1)}(D) &= D & g_2^{(1)}(D) &= 1 + D \\ g_1^{(2)}(D) &= 1 + D & g_2^{(2)}(D) &= 1 \\ g_1^{(3)}(D) &= 0 & g_2^{(3)}(D) &= 1 + D \end{aligned} \tag{8.1–8}$$

From the transfer functions and the D transform of the input sequences we obtain the D transform of the three output sequences as

$$\begin{aligned} c^{(1)}(D) &= u^{(1)}(D)g_1^{(1)}(D) + u^{(2)}(D)g_2^{(1)}(D) \\ c^{(2)}(D) &= u^{(1)}(D)g_1^{(2)}(D) + u^{(2)}(D)g_2^{(2)}(D) \\ c^{(3)}(D) &= u^{(1)}(D)g_1^{(3)}(D) + u^{(2)}(D)g_2^{(3)}(D) \end{aligned} \tag{8.1–9}$$

and finally

$$c(D) = c^{(1)}(D^3) + Dc^{(2)}(D^3) + D^2c^{(3)}(D^3) \tag{8.1–10}$$

Equation 8.1–9 can be written in a more compact way by defining

$$\boldsymbol{u}(D) = [\,u^{(1)}(D) \quad u^{(2)}(D)\,] \tag{8.1–11}$$

and

$$\boldsymbol{G}(D) = \begin{bmatrix} g_1^{(1)}(D) & g_1^{(2)}(D) & g_1^{(3)}(D) \\ g_2^{(1)}(D) & g_2^{(2)}(D) & g_2^{(3)}(D) \end{bmatrix} \tag{8.1–12}$$

By these definitions Equation 8.1–9 can be written as

$$\boldsymbol{c}(D) = \boldsymbol{u}(D)\boldsymbol{G}(D) \tag{8.1–13}$$

where

$$\boldsymbol{c}(D) = [\,c^{(1)}(D) \quad c^{(2)}(D) \quad c^{(3)}(D)\,] \tag{8.1–14}$$

In general, matrix $\boldsymbol{G}(D)$ is a $k \times n$ matrix whose elements are polynomials in D with degree at most $K - 1$. This matrix is called the *transform domain generator matrix* of the convolutional code. For the code whose encoder is shown in Figure 8.1–4 we have

$$G(D) = \begin{bmatrix} D & 1+D & 0 \\ 1+D & 1 & 1+D \end{bmatrix} \tag{8.1–15}$$

and for the convolutional code shown in Figure 8.1–2 we have

$$G(D) = [1 \quad D^2+1 \quad D^2+D+1] \tag{8.1–16}$$

8.1–1 Tree, Trellis, and State Diagrams

There are three alternative methods that are often used to describe a convolutional code. These are the *tree diagram*, the *trellis diagram*, and the *state diagram*. For example, the tree diagram for the convolutional encoder shown in Figure 8.1–2 is illustrated in Figure 8.1–5. Assuming that the encoder is in the all-zeros state initially, the diagram shows that if the first input bit is a 0, the output sequence is 000 and if the first bit is a 1, the output sequence is 111. Now, if the first input bit is a 1 and the second bit is a 0, the second set of 3 output bits is 001. Continuing through the tree, we see that if the third bit is a 0, then the output is 011, while if the third bit is a 1, then the output is 100. Given that a particular sequence has taken us to a particular node in the tree, the branching rule is to follow the upper branch if the next input bit is a 0 and the lower branch if the bit is a 1. Thus, we trace a particular path through the tree that is determined by the input sequence.

Close observation of the tree that is generated by the convolutional encoder shown in Figure 8.1–5 reveals that the structure repeats itself after the third stage. This behavior is consistent with the fact that the constraint length $K = 3$. That is, the 3-bit output sequence at each stage is determined by the input bit and the 2 previous input bits, i.e., the 2 bits contained in the first two stages of the shift register. The bit in the last stage of the shift register is shifted out at the right and does not affect the output. Thus we may say that the 3-bit output sequence for each input bit is determined by the input bit and the four possible states of the shift register, denoted as $a = 00, b = 01, c = 10, d = 11$.

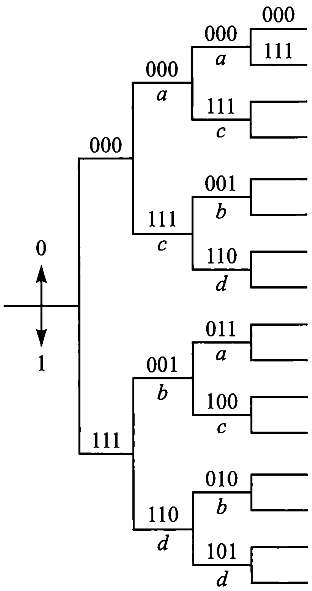


FIGURE 8.1–5
Tree diagram for rate 1/3, $K = 3$ convolutional code.

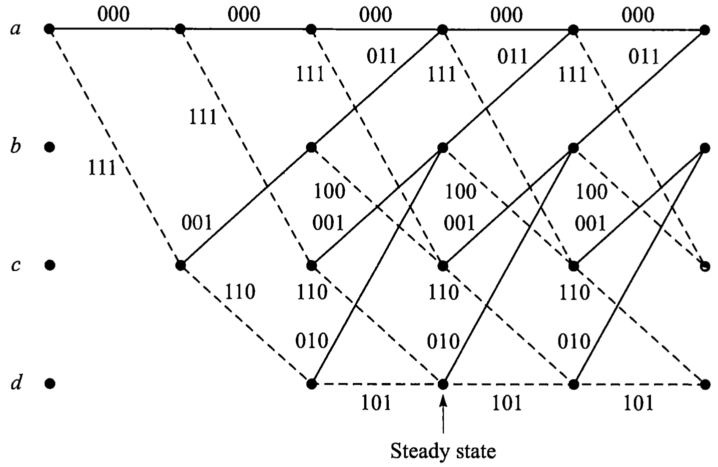


FIGURE 8.1–6
Trellis diagram for rate 1/3, $K = 3$ convolutional code.

If we label each node in the tree to correspond to the four possible states in the shift register, we find that at the third stage there are two nodes with label a, two with label b, two with label c, and two with label d. Now we observe that all branches emanating from two nodes having the same label (same state) are identical in the sense that they generate identical output sequences. This means that the two nodes having the same label can be merged. If we do this to the tree shown in Figure 8.1–5, we obtain another diagram, which is more compact, namely, a *trellis*. For example, the trellis diagram for the convolutional encoder of Figure 8.1–2 is shown in Figure 8.1–6. In drawing this diagram, we use the convention that a solid line denotes the output generated by the input bit 0 and a dotted line the output generated by the input bit 1. In the example being considered, we observe that, after the initial transient, the trellis contains four nodes at each stage, corresponding to the four states of the shift register, a, b, c, and d. After the second stage, each node in the trellis has two incoming paths and two outgoing paths. Of the two outgoing paths, one corresponds to the input bit 0 and the other to the path followed if the input bit is a 1.

Since the output of the encoder is determined by the input and the state of the encoder, an even more compact diagram than the trellis is the state diagram. The state diagram is simply a graph of the possible states of the encoder and the possible transitions from one state to another. For example, the state diagram for the encoder shown in Figure 8.1–2 is illustrated in Figure 8.1–7. This diagram shows that the possible transitions are

$$a \xrightarrow{0} a,\ a \xrightarrow{1} c,\ b \xrightarrow{0} a,\ b \xrightarrow{1} c,\ c \xrightarrow{0} b,\ c \xrightarrow{1} d,\ d \xrightarrow{0} b,\ d \xrightarrow{1} d$$

where $\alpha \xrightarrow{1} \beta$ denotes the transition from state α to β when the input bit is a 1. The 3 bits shown next to each branch in the state diagram represent the output bits. A dotted line in the graph indicates that the input bit is a 1, while the solid line indicates that the input bit is a 0.

EXAMPLE 8.1-3. Let us consider the $k = 2$, rate 2/3 convolutional code described in Example 8.1–2 and shown in Figure 8.1–3. The first two input bits may be 00, 01, 10,

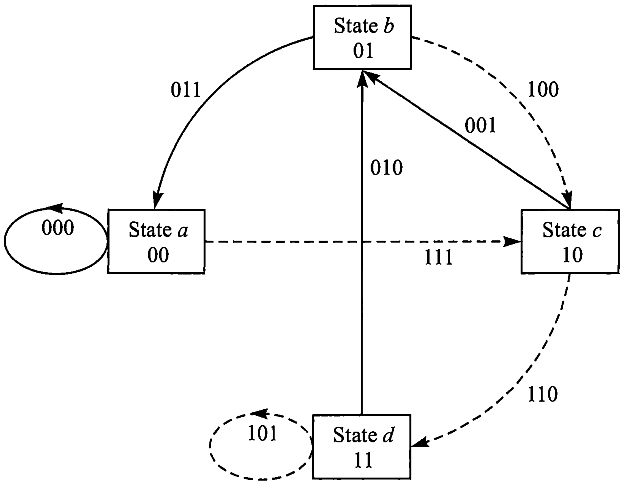

State b
01
011
100
001
010
000
State a
00
111
State c
10
110
101
State d
11

FIGURE 8.1–7
State diagram for rate 1/3, $K = 3$ convolutional code.

or 11. The corresponding output bits are 000, 010, 111, 101. When the next pair of input bits enters the encoder, the first pair is shifted to the second stage. The corresponding output bits depend on the pair of bits shifted into the second stage and the new pair of input bits. Hence, the tree diagram for this code, shown in Figure 8.1–8, has four branches per node, corresponding to the four possible pairs of input symbols.

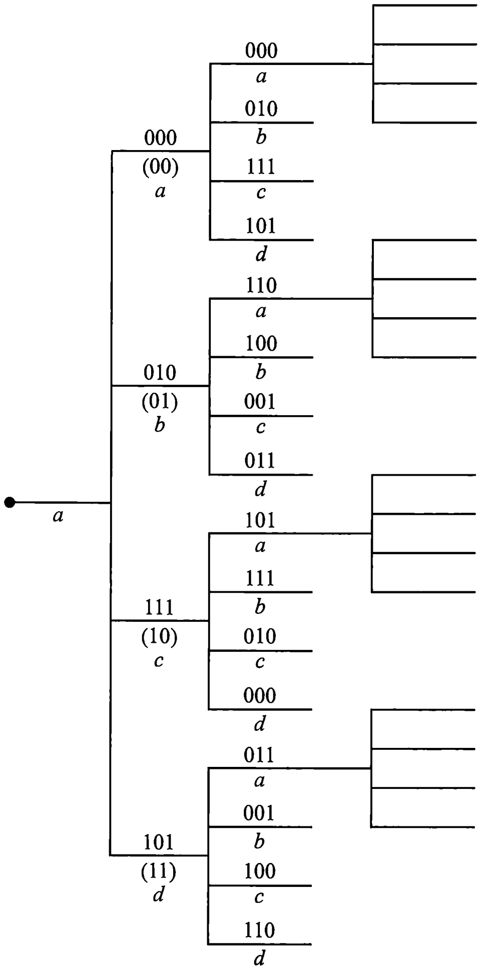

a
000
(00)
a
010
(01)
b
111
(10)
c
101
(11)
d
000
a
010
b
111
c
101
d
110
a
100
b
001
c
011
d
101
a
111
b
010
c
000
d
011
a
001
b
100
c
110
d

FIGURE 8.1–8
Tree diagram for $K = 2, k = 2, n = 3$ convolutional code.

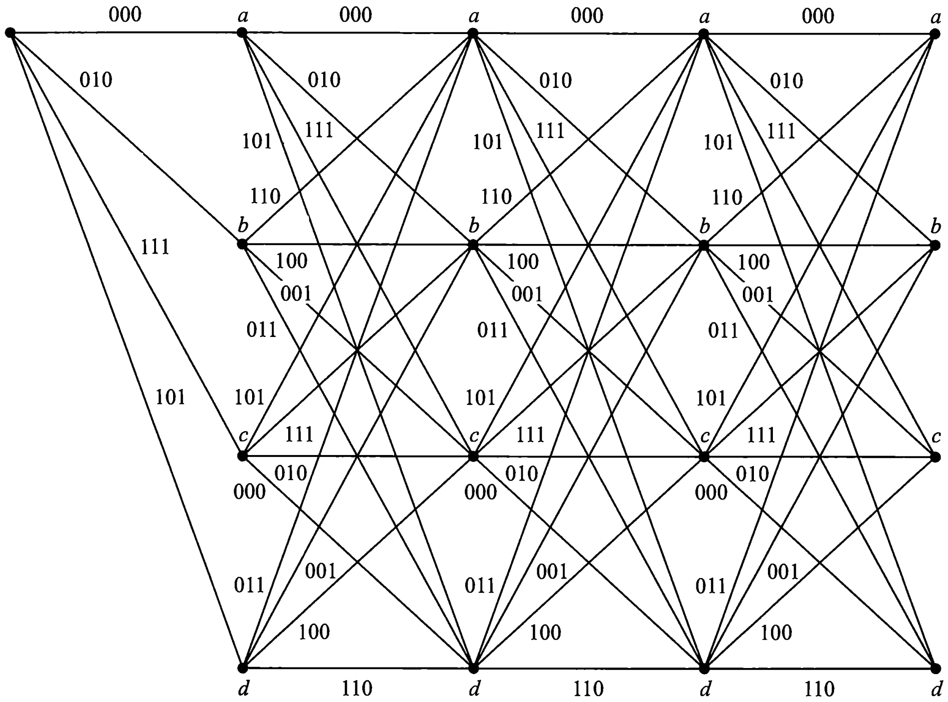


FIGURE 8.1–9
Trellis diagram for $K = 2, k = 2, n = 3$ convolutional code.

Since the constraint length of the code is $K = 2$, the tree begins to repeat after the second stage. As illustrated in Figure 8.1–8, all the branches emanating from nodes labeled a (state a) yield identical outputs.

By merging the nodes having identical labels, we obtain the trellis, which is shown in Figure 8.1–9. Finally, the state diagram for this code is shown in Figure 8.1–10.

To generalize, we state that a rate k/n, constraint length K, convolutional code is characterized by 2^k branches emanating from each node of the tree diagram. The trellis and the state diagrams each have $2^{k(K-1)}$ possible states. There are 2^k branches entering each state and 2^k branches leaving each state (in the trellis and tree, this is true after the initial transient). The three types of diagrams described above are also used to represent nonbinary convolutional codes. When the number of symbols in the code alphabet is $q = 2^k$, $k > 1$, the resulting nonbinary code may also be represented as an equivalent binary code. The following example considers a convolutional code of this type.

EXAMPLE 8.1–4. Let us consider the convolutional code generated by the encoder shown in Figure 8.1–11. This code may be described as a binary convolutional code with parameters $K = 2, k = 2, n = 4, R_c = 1/2$ and having the generators

$$g_1 = [1010], \quad g_2 = [0101], \quad g_3 = [1110], \quad g_4 = [1001]$$

Except for the difference in rate, this code is similar in form to the rate 2/3, $k = 2$ convolutional code considered in Example 8.1–2. Alternatively, the code generated by the encoder in Figure 8.1–11 may be described as a nonbinary ($q = 4$) code with one quaternary symbol as an input and two quaternary symbols as an output. In fact, if the output of the encoder is treated by the modulator and demodulator as q-ary ($q = 4$)

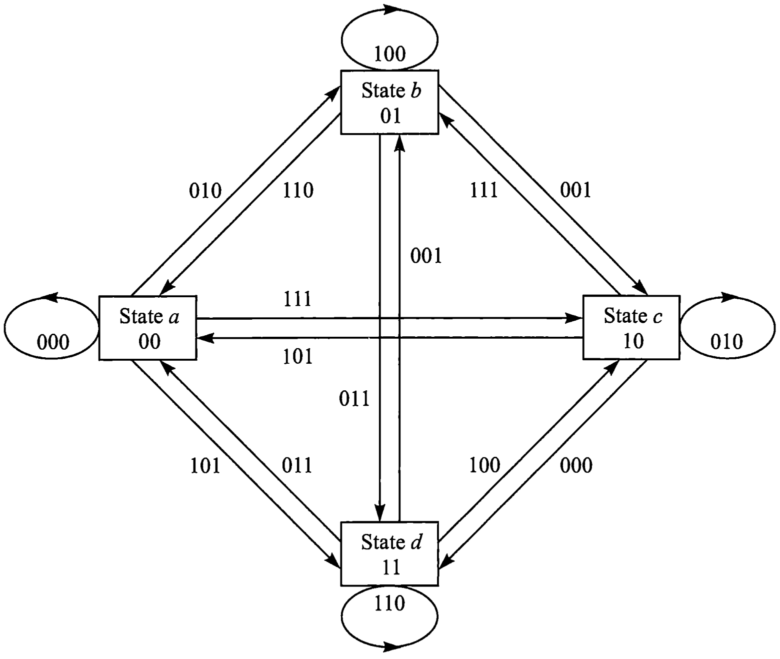

100
State b
01
010
110
111
001
001
State a
00
000
111
101
State c
10
010
011
101
011
100
000
State d
11
110

FIGURE 8.1–10
State diagram for $K = 2, k = 2, n = 3$ convolutional code.

symbols that are transmitted over the channel by means of some M-ary $(M = 4)$ modulation technique, the code is appropriately viewed as nonbinary. In any case, the tree, the trellis, and the state diagrams are independent of how we view the code. That is, this particular code is characterized by a tree with four branches emanating from each node, or a trellis with four possible states and four branches entering and leaving each state, or, equivalently, by a state diagram having the same parameters as the trellis.

8.1–2 The Transfer Function of a Convolutional Code

We have seen in Section 7.2–3 that the distance properties of block codes can be expressed in terms of the weight distribution, or weight enumeration polynomial of

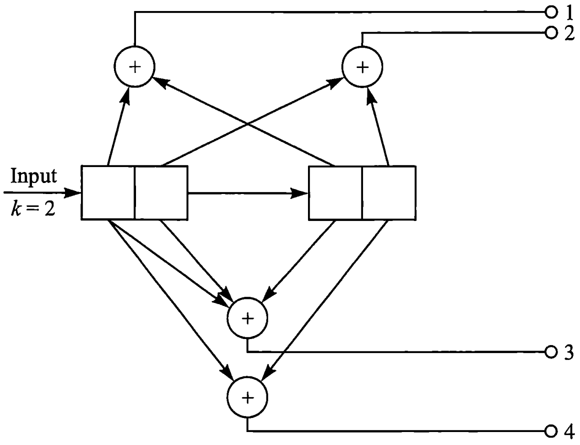

1
2
3
4
Input
k=2
+
+
+
+

FIGURE 8.1–11
$K = 2, k = 2, n = 4$ convolutional encoder.

the code. The weight distribution polynomial can be used to find performance bounds for linear block codes as given by Equations 7.2–39, 7.2–48, 7.4–4, and 7.5–17. The distance properties and the error rate performance of a convolutional code can be similarly obtained from its state diagram. Since a convolutional code is linear, the set of Hamming distances of the code sequences generated up to some stage in the tree, from the all-zero code sequence, is the same as the set of distances of the code sequences with respect to any other code sequence. Consequently, we assume without loss of generality that the all-zero code sequence is the input to the encoder. Therefore, instead of studying distance properties of the code we will study the weight distribution of the code, as we did for the case of block codes.

The state diagram shown in Figure 8.1–7 will be used to demonstrate the method for obtaining the distance properties of a convolutional code. We assume that the all-zero sequence is transmitted, and we focus on error events corresponding to a departure from the all-zero path on the code trellis and returning to it for the first time.

First, we label the branches of the state diagram as $Z^0 = 1$, Z^1, Z^2, or Z^3, where the exponent of Z denotes the Hamming distance between the sequence of output bits corresponding to each branch and the sequence of output bits corresponding to the all-zero branch. The self-loop at node a can be eliminated, since it contributes nothing to the distance properties of a code sequence relative to the all-zero code sequence and does not represent a departure from the all-zero sequence. Furthermore, node a is split into two nodes, one of which represents the input and the other the output of the state diagram, corresponding to the departure from the all-zero path and returning to it for the first time. Figure 8.1–12 illustrates the resulting diagram. We use this diagram, which now consists of five nodes because node a was split into two, to write the four state equations

$$\begin{aligned} X_c &= Z^3 X_a + Z X_b \\ X_b &= Z X_c + Z X_d \\ X_d &= Z^2 X_c + Z^2 X_d \\ X_e &= Z^2 X_b \end{aligned} \tag{8.1–17}$$

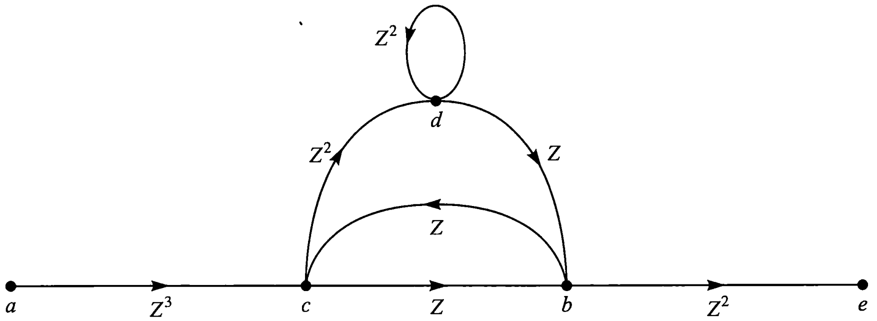


FIGURE 8.1–12
State diagram for rate $1/3$, $K = 3$ convolutional code.

The transfer function for the code is defined as $T(Z) = X_e/X_a$. By solving the state equations given above, we obtain

$$\begin{aligned} T(Z) &= \frac{Z^6}{1-2Z^2} \\ &= Z^6 + 2Z^8 + 4Z^{10} + 8Z^{12} + \cdots \\ &= \sum_{d=6}^{\infty} a_d Z^d \end{aligned} \tag{8.1–18}$$

where, by definition,

$$a_d = \begin{cases} 2^{(d-6)/2} & \text{even } d \\ 0 & \text{odd } d \end{cases} \tag{8.1–19}$$

The transfer function for this code indicates that there is a single path of Hamming distance $d = 6$ from the all-zero path that merges with the all-zero path at a given node. From the state diagram shown in Figure 8.1–7 or the trellis diagram shown in Figure 8.1–6, it is observed that the $d = 6$ path is *acbe*. There is no other path from node a to node e having a distance $d = 6$. The second term in Equation 8.1–18 indicates that there are two paths from node a to node e having a distance $d = 8$. Again, from the state diagram or the trellis, we observe that these paths are *acdbe* and *acbcbe*. The third term in Equation 8.1–18 indicates that there are four paths of distance $d = 10$, and so forth. Thus the transfer function gives us the distance properties of the convolutional code. The minimum distance of the code is called the *minimum free distance* and denoted by d_{free}. In our example, $d_{\text{free}} = 6$.

The transfer function $T(Z)$ introduced above is similar to the the weight enumeration function (WEF) $A(Z)$ for block codes introduced in Chapter 7. The main difference is that in the transfer function of a convolutional code the term corresponding to the loop at the all-zero state is eliminated; hence the all-zero code sequence is not included, and therefore the lowest power in the transfer function is d_{free}. In determining $A(Z)$ we include the all-zero codeword, hence $A(Z)$ always contains a constant equal to 1. Another difference is that in determining the transfer function of a convolutional code, we consider only paths in the trellis that depart from the all-zero state and return to it for the *first time*. Such a path is called a *first event error* and is used to bound the error probability of convolutional codes.

The transfer function can be used to provide more detailed information than just the distance of the various paths. Suppose we introduce a factor Y into all branch transitions caused by the input bit 1. Thus, as each branch is traversed, the cumulative exponent on Y increases by 1 only if that branch transition is due to an input bit 1. Furthermore, we introduce a factor of J into each branch of the state diagram so that the exponent of J will serve as a counting variable to indicate the number of branches in any given path from node a to node e. For the rate 1/3 convolutional code in our example, the state diagram that incorporates the additional factors of J and Y is shown in Figure 8.1–13.

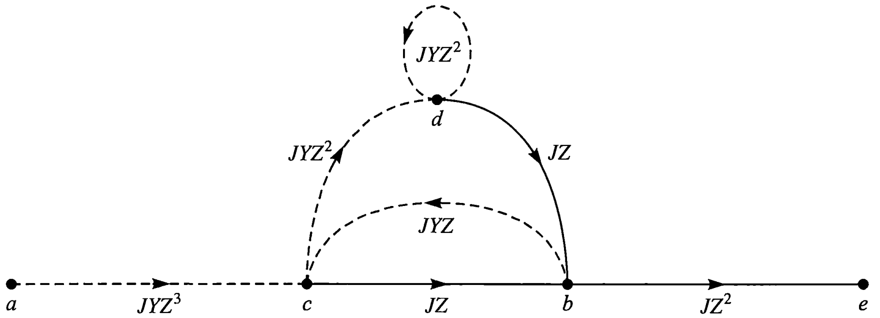


FIGURE 8.1–13
State diagram for rate 1/3, $K = 3$ convolutional code.

The state equations for the state diagram shown in Figure 8.1–13 are

$$\begin{aligned} X_c &= JYZ^3X_a + JYZX_b \\ X_b &= JZX_c + JZX_d \\ X_d &= JYZ^2X_c + JYZ^2X_d \\ X_e &= JZ^2X_b \end{aligned} \tag{8.1–20}$$

Upon solving these equations for the ratio X_e/X_a, we obtain the transfer function

$$\begin{aligned} T(Y, Z, J) &= \frac{J^3YZ^6}{1 - JYZ^2(1+J)} \\ &= J^3YZ^6 + J^4Y^2Z^8 + J^5Y^2Z^8 + J^5Y^3Z^{10} \\ &\quad + 2J^6Y^3Z^{10} + J^7Y^3Z^{10} + \cdots \end{aligned} \tag{8.1–21}$$

This form for the transfer functions gives the properties of all the paths in the convolutional code. That is, the first term in the expansion of $T(Y, Z, J)$ indicates that the distance $d = 6$ path is of length 3 and of the three information bits, one is a 1. The second and third terms in the expansion of $T(Y, Z, J)$ indicate that of the two $d = 8$ terms, one is of length 4 and the second has length 5. Two of the four information bits in the path having length 4 and two of the five information bits in the path having length 5 are 1s. Thus, the exponent of the factor J indicates the length of the path that merges with the all-zero path for the first time, the exponent of the factor Y indicates the number of 1s in the information sequence for that path, and the exponent of Z indicates the distance of the sequence of encoded bits for that path from the all-zero sequence (the weight of the code sequence).

The factor J is particularly important if we are transmitting a sequence of finite duration, say m bits. In such a case, the convolutional code is truncated after m nodes or m branches. This implies that the transfer function for the truncated code is obtained by truncating $T(Y, Z, J)$ at the term J^m. On the other hand, if we are transmitting an extremely long sequence, i.e., essentially an infinite-length sequence, we may wish to suppress the dependence of $T(Y, Z, J)$ on the parameter J. This is easily accomplished by setting $J = 1$. Hence, for the example given above, we have

$$\begin{aligned} T(Y, Z) = T(Y, Z, 1) &= \frac{YZ^6}{1 - 2YZ^2} \\ &= YZ^6 + 2Y^2Z^8 + 4Y^3Z^{10} + \cdots \\ &= \sum_{d=6}^{\infty} a_d Y^{(d-4)/2} Z^d \end{aligned} \tag{8.1–22}$$

where the coefficients $\{a_d\}$ are defined by Equation 8.1–19. The reader should note the similarity between $T(Y, Z)$ and $B(Y, Z)$ introduced in Equation 7.2–25, Section 7.2–3.

The procedure outlined above for determining the transfer function of a binary convolutional code can be applied easily to simple codes with few number of states. For a general procedure for finding the transfer function of a convolutional code based on application of Mason's rule for deriving transfer function of flow graphs, the reader is referred to Lin and Costello (2004).

The procedure outlined above can be easily extended to nonbinary codes. In the following example, we determine the transfer function of the nonbinary convolutional code previously introduced in Example 8.1–4.

EXAMPLE 8.1-5. The convolutional code shown in Figure 8.1–11 has the parameters $K = 2, k = 2, n = 4$. In this example, we have a choice of how we label distances and count errors, depending on whether we treat the code as binary or nonbinary. Suppose we treat the code as nonbinary. Thus, the input to the encoder and the output are treated as quaternary symbols. In particular, if we treat the input and output as quaternary symbols 00, 01, 10, and 11, the distance measured in symbols between the sequences 0111 and 0000 is 2. Furthermore, suppose that an input symbol 00 is decoded as the symbol 11; then we have made one symbol error. This convention applied to the convolutional code shown in Figure 8.1–11 results in the state diagram illustrated in Figure 8.1–14, from which we obtain the state equations

$$\begin{aligned} X_b &= YJZ^2X_a + YJZX_b + YJZX_c + YJZ^2X_d \\ X_c &= YJZ^2X_a + YJZ^2X_b + YJZX_c + YJZX_d \\ X_d &= YJZ^2X_a + YJZX_b + YJZ^2X_c + YJZX_d \\ X_c &= JZ^2(X_b + X_c + X_d) \end{aligned} \tag{8.1–23}$$

Solution of these equations leads to the transfer function

$$T(Y, Z, J) = \frac{3YJ^2Z^4}{1 - 2YJZ - YJZ^2} \tag{8.1–24}$$

This expression for the transfer function is particularly appropriate when the quaternary symbols at the output of the encoder are mapped into a corresponding set of quaternary waveforms $s_m(t)$, $m = 1, 2, 3, 4$, e.g., four orthogonal waveforms. Thus, there is a one-to-one correspondence between code symbols and signal waveforms. Alternatively, for example, the output of the encoder may be transmitted as a sequence of binary digits by means of binary PSK. In such a case, it is appropriate to measure distance in terms

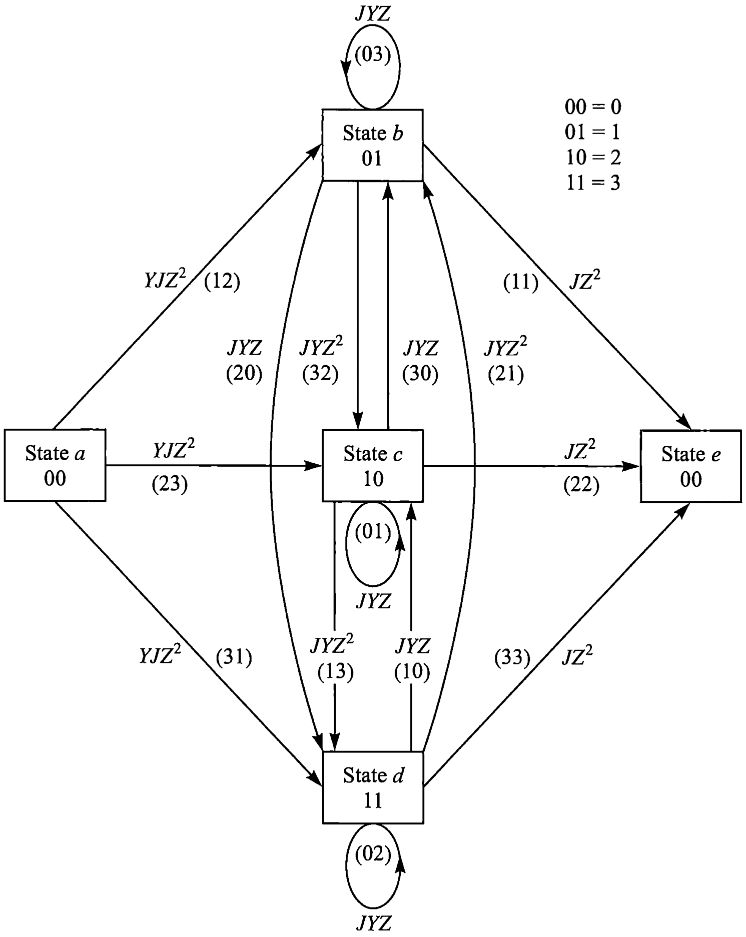

JYZ
(03)
State b
01
00 = 0
01 = 1
10 = 2
11 = 3
YJZ²
(12)
(11)
JZ²
JYZ
(20)
JYZ²
(32)
JYZ
(30)
JYZ²
(21)
State a
00
YJZ²
(23)
State c
10
JZ²
(22)
State e
00
(01)
JYZ
YJZ²
(31)
JYZ²
(13)
JYZ
(10)
(33)
JZ²
State d
11
(02)
JYZ

FIGURE 8.1–14
State diagram for $K = 2, k = 2$, rate 1/2 nonbinary code.

of bits. When this convention is employed, the state diagram is labeled as shown in Figure 8.1–15. Solution of the state equations obtained from this state diagram yields a transfer function that is different from the one given in Equation 8.1–9.

8.1–3 Systematic, Nonrecursive, and Recursive Convolutional Codes

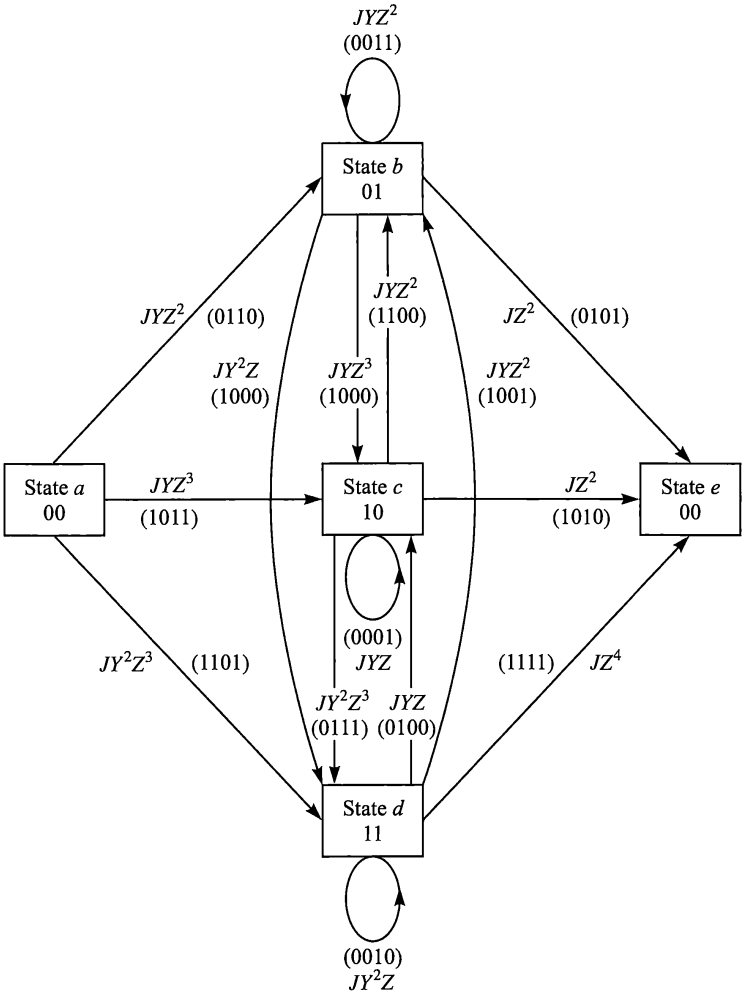
A convolutional code in which the information sequence directly appears as part of the code sequence is called *systematic*. For instance the convolutional encoder given in Figure 8.1–2 depicts the encoder for a systematic convolutional code since

$$c^{(1)} = \boldsymbol{u} \star \boldsymbol{g}_1 = \boldsymbol{u} \qquad (8.1\text{–}25)$$

This shows that the information sequence $\boldsymbol{u}$ appears as part of the code sequence $\boldsymbol{c}$. This can be directly seen by observing that the transform domain generator matrix of the code given in Equation 8.1–16 has a 1 in its first column.

In general, if $\boldsymbol{G}(D)$ is of the form

$$\boldsymbol{G}(D) = [\,\boldsymbol{I}_k \mid \boldsymbol{P}(D)\,] \qquad (8.1\text{–}26)$$



JYZ^2 (0011)
State b 01
JYZ^2 (0110)
JY^2Z (1000)
JYZ^2 (1100)
JYZ^3 (1000)
JZ^2 (0101)
JYZ^2 (1001)
State a 00
JYZ^3 (1011)
State c 10
JZ^2 (1010)
State e 00
(0001) JYZ
JY^2Z^3 (1101)
JY^2Z^3 (0111)
JYZ (0100)
(1111) JZ^4
State d 11
(0010) JY^2Z

FIGURE 8.1–15

State diagram for $K = 2$, $k = 2$, rate 1/2 convolutional code with output treated as a binary sequence.

where $\boldsymbol{P}(D)$ is a $k \times (n - k)$ polynomial matrix, the convolutional code is systematic. The matrix $\boldsymbol{G}(D)$ given below corresponds to a systematic convolutional code with $n = 3$ and $k = 2$.

$$\boldsymbol{G}(D) = \begin{bmatrix} 1 & 0 & 1+D \\ 0 & 1 & 1+D+D^2 \end{bmatrix} \tag{8.1–27}$$

Two convolutional encoders are called *equivalent* if the code sequences generated by them are the same. Note that in the definition of equivalent convolutional encoders it is sufficient that the code sequences be the same; it is not required that the equal code sequences correspond to the same information sequences.

EXAMPLE 8.1–6. A convolutional code with $n = 3$ and $k = 1$ is described by

$$\boldsymbol{G}(D) = [1 + D + D^2 \quad 1 + D \quad D] \tag{8.1–28}$$

The code sequences generated by this encoder are sequences of the general form

$$c(D) = c^{(1)}(D^3) + Dc^{(2)}(D^3) + D^2c^{(3)}(D^3) \tag{8.1–29}$$

where

$$\begin{aligned} c^{(1)}(D) &= (1 + D + D^2)u(D) \\ c^{(2)}(D) &= (1 + D)u(D) \\ c^{(3)}(D) &= Du(D) \end{aligned} \tag{8.1–30}$$

or

$$c(D) = (1 + D + D^3 + D^4 + D^5 + D^6)u(D^3) \tag{8.1–31}$$

The matrix $\boldsymbol{G}(D)$ can also be written as

$$\begin{aligned} \boldsymbol{G}(D) &= (1 + D + D^2)\begin{bmatrix} 1 & \frac{1+D}{1+D+D^2} & \frac{D}{1+D+D^2} \end{bmatrix} \\ &= (1 + D + D^2)\boldsymbol{G}'(D) \end{aligned} \tag{8.1–32}$$

$\boldsymbol{G}(D)$ and $\boldsymbol{G}'(D)$ are equivalent encoders, meaning that these two matrices generate the same set of code sequences; However, these code sequences correspond to different information sequences. Also note that $\boldsymbol{G}'(D)$ represents a systematic convolutional code.

It is easy to verify that the information sequences $\boldsymbol{u} = (1, 0, 0, 0, 0, \dots)$ and $\boldsymbol{u}' = (1, 1, 1, 0, 0, 0, 0, \dots)$ when applied to encoders $\boldsymbol{G}(D)$ and $\boldsymbol{G}'(D)$, respectively, generate the same code sequence

$$\boldsymbol{c} = (1, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, \dots)$$

The transform domain generator matrix $\boldsymbol{G}'(D)$ given by

$$\boldsymbol{G}'(D) = \begin{bmatrix} 1 & \frac{1+D}{1+D+D^2} & \frac{D}{1+D+D^2} \end{bmatrix} \tag{8.1–33}$$

represents a convolutional encoder with feedback. To realize this transfer function, we need to use shift registers with feedback as shown in Figure 8.1–16.

Convolutional codes that are realized using feedback shift registers are called *recursive convolutional codes* (RCCs). The transform domain generator matrix for these codes includes ratios of polynomials whereas in the case of nonrecursive convolutional codes the elements of $\boldsymbol{G}(D)$ are polynomials. Note that in recursive convolutional codes the existence of feedback causes the code to have infinite-length impulse responses.

Although systematic convolutional codes are desirable, unfortunately, in general systematic nonrecursive convolutional codes cannot achieve the highest free distance possible with nonsystematic nonrecursive convolutional codes of the same rate and constraint length. Recursive systematic convolutional codes, however, can achieve the

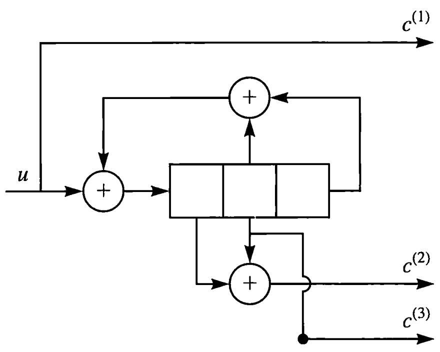


FIGURE 8.1–16
Realization of $\boldsymbol{G}'(D)$ using feedback shift register.

same free distance as nonrecursive systematic codes for a given rate and constraint length. The code depicted in Figure 8.1–16 is a *recursive systematic convolutional code* (RSCC). Such codes are essential parts of turbo codes as discussed in Section 8.9.

8.1–4 The Inverse of a Convolutional Encoder and Catastrophic Codes

One desirable property of a convolutional encoder is that in the absence of noise it is possible to recover the information sequence from the encoded sequence. In other words it is desirable that the encoding process be *invertible*. Clearly, any systematic convolutional code is invertible.

In addition to invertibility, it is desirable that the inverse of the encoder be realizable using a feedforward network. The reason is that if in transmission of $c(D)$ one error occurs and the inverse function is a feedback circuit having an infinite impulse response, then this single error, which is equivalent to an impulse, causes an infinite number of errors to occur at the output.

For a nonsystematic convolutional code, there exists a one-to-one correspondence between $c(D)$ and $c^{(1)}(D), c^{(2)}(D), \ldots, c^{(n)}(D)$ and also between $u(D)$ and $u^{(1)}(D), u^{(2)}(D), \ldots, u^{(k)}(D)$. Therefore, to be able to recover $u(D)$ from $c(D)$, we have to be able to recover $u^{(1)}(D), u^{(2)}(D), \ldots, u^{(k)}(D)$ from $c^{(1)}(D), c^{(2)}(D), \ldots, c^{(n)}(D)$. Using the relation

$$\boldsymbol{c}(D) = \boldsymbol{u}(D)\boldsymbol{G}(D) \tag{8.1–34}$$

we conclude that the code is invertible if $\boldsymbol{G}(D)$ is invertible. Therefore the condition for invertibility of a convolutional code is that for the $k \times n$ matrix $\boldsymbol{G}(D)$ there must exist an $n \times k$ inverse matrix $\boldsymbol{G}^{-1}(D)$ such that

$$\boldsymbol{G}(D)\boldsymbol{G}^{-1}(D) = D^l \boldsymbol{I}_k \tag{8.1–35}$$

where $l \geq 0$ is an integer representing a delay of l time units between the input and the output.

The following result due to Massey and Sain (1968) provides the necessary and sufficient condition under which a feedforward inverse for $\boldsymbol{G}(D)$ exists.

An (n, k) convolutional code with

$$\boldsymbol{G}(D) = [\,\boldsymbol{g}_1(D) \quad \boldsymbol{g}_2(D) \quad \cdots \quad \boldsymbol{g}_n(D)\,] \tag{8.1–36}$$

has a feedforward inverse with delay l if and only if for some $l \geq 0$ we have

$$\mathrm{GCD}\,\{\boldsymbol{g}_i(D), 1 \leq i \leq k\} = D^l \tag{8.1–37}$$

where GCD denotes the greatest common divisor. For (n, k) convolutional codes the condition is

$$\mathrm{GCD}\left\{\Delta_i(D), 1 \leq i \leq \binom{n}{k}\right\} = D^l \tag{8.1–38}$$

where $\Delta_i(D)$, $1 \leq i \leq \binom{n}{k}$ denote the determinants of the $\binom{n}{k}$ distinct $k \times k$ submatrices of $\boldsymbol{G}(D)$.

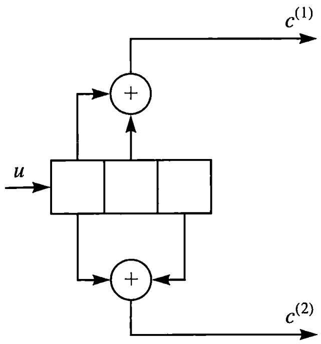


FIGURE 8.1–17
A catastrophic convolutional encoder.

Convolutional codes for which a feedforward inverse does not exist are called *catastrophic convolutional codes*. When a catastrophic convolutional code is used on a binary symmetric channel, it is possible for a finite number of channel errors to cause an infinite number of decoding errors. For simple codes, such a code can be identified from its state diagram. It will contain a zero-distance path (a path with multiplier $D^0 = 1$) from some nonzero state back to the same state. This means that one can loop around this zero-distance path an infinite number of times without increasing the distance relative to the all-zero path. But, if this self-loop corresponds to the transmission of a 1, the decoder will make an infinite number of errors. For general convolutional codes, conditions given in Equations 8.1–37 and 8.1–38 must be satisfied for the code to be noncatastrophic.

EXAMPLE 8.1–7. Consider the $k = 1, n = 2, K = 3$ convolutional code shown in Figure 8.1–17. For this code $G(D)$ is given by

$$G(D) = [\,1 + D \quad 1 + D^2\,] \tag{8.1–39}$$

and since $\mathrm{GCD}\{1 + D, 1 + D^2\} = 1 + D \neq D^l$, the code is catastrophic. The state diagram for this code is shown in Figure 8.1–18. The existence of the self-loop from state 11 to itself corresponding to an input sequence of weight 1 and output sequence of weight 0 results in catastrophic behavior for this code.

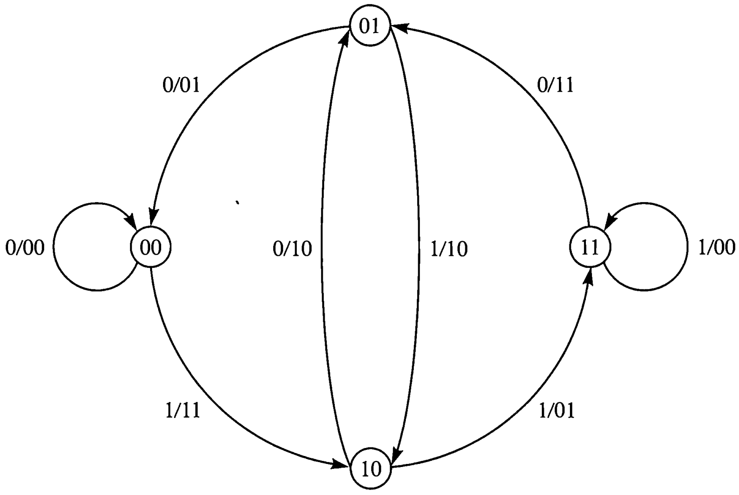


FIGURE 8.1–18
The state diagram for the catastrophic code of Figure 8.1–17.

8.2
DECODING OF CONVOLUTIONAL CODES

There exist different methods for decoding of convolutional codes. Similar to block codes, the decoding of convolutional codes can be done either by soft decision or by hard decision decoding. In addition, the optimal decoding of convolutional codes can employ the maximum-likelihood or the maximum a posteriori principle. For convolutional codes with high constraint lengths, optimal decoding algorithms become too complex. Suboptimal decoding algorithms are usually used in such cases.

8.2–1 Maximum-Likelihood Decoding of Convolutional Codes — The Viterbi Algorithm

In the decoding of a block code for a memoryless channel, we computed the distances (Hamming distance for hard-decision decoding and Euclidean distance for soft-decision decoding) between the received codeword and the 2^k possible transmitted codewords. Then we selected the codeword that was closest in distance to the received codeword. This decision rule, which requires the computation of 2^k metrics, is optimum in the sense that it results in a minimum probability of error for the binary symmetric channel with $p < \frac{1}{2}$ and the additive white Gaussian noise channel.

Unlike a block code, which has a fixed length n, a convolutional encoder is basically a finite-state machine. Hence the optimum decoder is a maximum-likelihood sequence estimator (MLSE) of the type described in Section 4.8–1 for signals with memory. Therefore, optimum decoding of a convolutional code involves a search through the trellis for the most probable sequence. Depending on whether the detector following the demodulator performs hard or soft decisions, the corresponding metric in the trellis search may be either a Hamming metric or a Euclidean metric, respectively. We elaborate below, using the trellis in Figure 8.1–6 for the convolutional code shown in Figure 8.1–2.

Consider the two paths in the trellis that begin at the initial state a and remerge at state a after three state transitions (three branches), corresponding to the two information sequences 000 and 100 and the transmitted sequences 000 000 000 and 111 001 011, respectively. We denote the transmitted bits by $\{c_{jm}, j = 1, 2, 3; m = 1, 2, 3\}$, where the index j indicates the jth branch and the index m the mth bit in that branch. Correspondingly, we define $\{r_{jm}, j = 1, 2, 3; m = 1, 2, 3\}$ as the output of the demodulator. If the decoder performs hard decision decoding, the detector output for each transmitted bit is either 0 or 1. On the other hand, if soft decision decoding is employed and the coded sequence is transmitted by binary coherent PSK, the input to the decoder is

$$r_{jm} = \sqrt{\mathcal{E}_c}(2c_{jm} - 1) + n_{jm} \tag{8.2–1}$$

where n_{jm} represents the additive noise and $\mathcal{E}_c$ is the transmitted signal energy for each code bit.

A metric is defined for the jth branch of the ith path through the trellis as the logarithm of the joint probability of the sequence $\{r_{jm}, m = 1, 2, 3\}$ conditioned on the transmitted sequence $\{c_{jm}^{(i)}, m = 1, 2, 3\}$ for the ith path. That is,

$$\mu_j^{(i)} = \log p\big(\boldsymbol{r}_j | \boldsymbol{c}_j^{(i)}\big), \qquad j = 1, 2, 3, \ldots \tag{8.2–2}$$

Furthermore, a metric for the ith path consisting of B branches through the trellis is defined as

$$PM^{(i)} = \sum_{j=1}^{B} \mu_j^{(i)} \tag{8.2–3}$$

The criterion for deciding between two paths through the trellis is to select the one having the larger metric. This rule maximizes the probability of a correct decision, or, equivalently, it minimizes the probability of error for the sequence of information bits. For example, suppose that hard decision decoding is performed by the demodulator, yielding the received sequence $\{101\ 000\ 100\}$. Let $i = 0$ denote the three-branch all-zero path and $i = 1$ the second three-branch path that begins in the initial state a and remerges with the all-zero path at state a after three transitions. The metrics for these two paths are

$$\begin{aligned} PM^{(0)} &= 6\log(1-p) + 3\log p \\ PM^{(1)} &= 4\log(1-p) + 5\log p \end{aligned} \tag{8.2–4}$$

where p is the probability of a bit error. Assuming that $p < \frac{1}{2}$, we find that the metric $PM^{(0)}$ is larger than the metric $PM^{(1)}$. This result is consistent with the observation that the all-zero path is at Hamming distance $d = 3$ from the received sequence, while the $i = 1$ path is at Hamming distance $d = 5$ from the received path. Thus, the Hamming distance is an equivalent metric for hard decision decoding.

Similarly, suppose that soft decision decoding is employed and the channel adds white Gaussian noise to the signal. Then the demodulator output is described statistically by the probability density function

$$p\big(r_{jm} | c_{jm}^{(i)}\big) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{\left[r_{jmc} - \sqrt{\mathcal{E}}\big(2c_{jm}^{(i)} - 1\big)\right]^2}{2\sigma^2} \right\} \tag{8.2–5}$$

where $\sigma^2 = \frac{1}{2}N_0$ is the variance of the additive Gaussian noise. If we neglect the terms that are common to all branch metrics, the branch metric for the jth branch of the ith path may be expressed as

$$\mu_j^{(i)} = \sum_{m=1}^{n} r_{jm}\big(2c_{jm}^{(i)} - 1\big) \tag{8.2–6}$$

where, in our example, $n = 3$. Thus the correlation metrics for the two paths under consideration are

$$\begin{aligned} CM^{(0)} &= \sum_{j=1}^{3} \sum_{m=1}^{3} r_{jm} \left(2c_{jm}^{(0)} - 1\right) \\ CM^{(1)} &= \sum_{j=1}^{3} \sum_{m=1}^{3} r_{jm} \left(2c_{jm}^{(1)} - 1\right) \end{aligned} \tag{8.2–7}$$

From the above discussion it is observed that for ML decoding we need to look for a code sequence $\boldsymbol{c}^{(m)}$ in the trellis $\mathcal{T}$ that satisfies

$$\begin{aligned} \boldsymbol{c}^{(m)} &= \max_{\boldsymbol{c} \in \mathcal{T}} \sum_{j} \log p(\boldsymbol{r}_j | \boldsymbol{c}_j), && \text{for a general memoryless channel} \\ \boldsymbol{c}^{(m)} &= \min_{\boldsymbol{c} \in \mathcal{T}} \sum_{j} \left\| \boldsymbol{r}_j - \boldsymbol{c}_j \right\|^2, && \text{for soft decision decoding} \\ \boldsymbol{c}^{(m)} &= \min_{\boldsymbol{c} \in \mathcal{T}} \sum_{j} d_{\mathrm{H}}(\boldsymbol{y}_j, \boldsymbol{c}_j), && \text{for hard decision decoding} \end{aligned} \tag{8.2–8}$$

Note that for hard decision decoding $\boldsymbol{y}$ denotes the result of binary (hard) decisions on the demodulator output $\boldsymbol{r}$. Also in the hard decision case, $\boldsymbol{c}$ denotes the binary encoded sequence whose components are 0 and 1, whereas in the soft decision case the components of $\boldsymbol{c}$ are $\pm\sqrt{\mathcal{E}_c}$. What is clear from above is that in all cases maximum-likelihood decoding requires finding a path in the trellis that minimizes or maximizes an additive metric. This is done by using the Viterbi algorithm as discussed below.

We consider the two paths described above, which merge at state a after three transitions. Note that any particular path through the trellis that stems from this node will add identical terms to the path metrics $CM^{(0)}$ and $CM^{(1)}$. Consequently, if $CM^{(0)} > CM^{(1)}$ at the merged node a after three transitions, $CM^{(0)}$ will continue to be larger than $CM^{(1)}$ for any path that stems from node a. This means that the path corresponding to $CM^{(1)}$ can be discarded from further consideration. The path corresponding to the metric $CM^{(0)}$ is the *survivor*. Similarly, one of the two paths that merge at state b can be eliminated on the basis of the two corresponding metrics. This procedure is repeated at state c and state d. As a result, after the first three transitions, there are four surviving paths, one terminating at each state, and a corresponding metric for each survivor. This procedure is repeated at each stage of the trellis as new signals are received in subsequent time intervals.

In general, when a binary convolutional code with $k = 1$ and constraint length K is decoded by means of the Viterbi algorithm, there are 2^{K-1} states. Hence, there are 2^{K-1} surviving paths at each stage and 2^{K-1} metrics, one for each surviving path. Furthermore, a binary convolutional code in which k bits at a time are shifted into an encoder that consists of K (k-bit) shift-register stages generates a trellis that has $2^{k(K-1)}$ states. Consequently, the decoding of such a code by means of the Viterbi algorithm requires keeping track of $2^{k(K-1)}$ surviving paths and $2^{k(K-1)}$ metrics. At each stage of the trellis, there are 2^k paths that merge at each node. Since each path that converges at a common node requires the computation of a metric, there are

2^k metrics computed for each node. Of the 2^k paths that merge at each node, only one survives, and this is the most probable (minimum-distance) path. Thus, the number of computations in decoding performed at each stage increases exponentially with k and K. The exponential increase in computational burden limits the use of the Viterbi algorithm to relatively small values of K and k.

The decoding delay in decoding a long information sequence that has been convolutionally encoded is usually too long for most practical applications. Moreover, the memory required to store the entire length of surviving sequences is large and expensive. As indicated in Section 4.8–1, a solution to this problem is to modify the Viterbi algorithm in a way which results in a fixed decoding delay without significantly affecting the optimal performance of the algorithm. Recall that the modification is to retain at any given time t only the most recent δ decoded information bits (symbols) in each surviving sequence. As each new information bit (symbol) is received, a final decision is made on the bit (symbol) received δ branches back in the trellis, by comparing the metrics in the surviving sequences and deciding in favor of the bit in the sequence having the largest metric. If δ is chosen sufficiently large, all surviving sequences will contain the identical decoded bit (symbol) δ branches back in time. That is, with high probability, all surviving sequences at time t stem from the same node at $t - \delta$. It has been found experimentally (computer simulation) that a delay $\delta \geq 5K$ results in a negligible degradation in the performance relative to the optimum Viterbi algorithm.

8.2–2 Probability of Error for Maximum-Likelihood Decoding of Convolutional Codes

In deriving the probability of error for convolutional codes, the linearity property for this class of codes is employed to simplify the derivation. That is, we assume that the all-zero sequence is transmitted, and we determine the probability of error in deciding in favor of another sequence.

Since the convolutional code does not necessarily have a fixed length, we derive its performance from the probability of error for sequences that merge with the all-zero sequence for the first time at a given node in the trellis. In particular, we define the *first-event error probability* as the probability that another path that merges with the all-zero path at node B has a metric that exceeds the metric of the all-zero path for the first time. Of course in transmission of convolutional codes, other types of errors can occur; but it can be shown that bounding the error probability of the convolutional code by the sum of first-event error probabilities provides an upper bound that, although conservative, in most cases is a usable bound on the error probability. The interested user can refer to the book by Lin and Costello (2004) for details.

As we have previously discussed in Section 8.1–2, the transfer function of a convolutional code is similar to the WEF of a block code with two differences. First, it considers only the first-event errors; and second, it does not include the all-zero code sequence. Therefore, parallel to the argument we presented for block codes in Section 7.2–4, we can derive bounds on sequence and bit error probability of convolutional codes.

The sequence error probability of a convolutional code is bounded by

$$P_e \leq T(Z)\Big|_{Z=\Delta} \tag{8.2–9}$$

where

$$\Delta = \sum_{y \in \mathscr{Y}} \sqrt{p(y|0)p(y|1)} \tag{8.2–10}$$

Note that unlike Equation 7.2–39, which states in linear block codes $P_e \leq A(\Delta) - 1$, here we do not need to subtract 1 from $T(Z)$ since $T(Z)$ does not include the all-zero path. Equation 8.2–9 can be written as

$$P_e \leq \sum_{d=d_{\text{free}}}^{\infty} a_d \Delta^d \tag{8.2–11}$$

The bit error probability for a convolutional code follows from Equation 7.2–48 as

$$P_b \leq \frac{1}{k} \frac{\partial}{\partial Y} T(Y, Z)\Big|_{Y=1, Z=\Delta} \tag{8.2–12}$$

From Example 6.8–1 we know that if the modulation is BPSK (or QPSK) and the channel is an AWGN channel with soft decision decoding, then

$$\Delta = e^{-R_c \gamma_b} \tag{8.2–13}$$

and in case of hard decision decoding, where the channel model is a binary symmetric channel with crossover probability of p, we have

$$\Delta = \sqrt{4p(1-p)} \tag{8.2–14}$$

Therefore, we have the following upper bounds for the bit error probability of a convolutional code:

$$P_b \leq \begin{cases} \frac{1}{k} \frac{\partial}{\partial Y} T(Y, Z)\Big|_{Y=1, Z=\exp(-R_c \gamma_b)} & \text{BPSK with soft decision decoding} \\ \frac{1}{k} \frac{\partial}{\partial Y} T(Y, Z)\Big|_{Y=1, Z=\sqrt{4p(1-p)}} & \text{hard decision decoding} \end{cases} \tag{8.2–15}$$

In hard decision decoding we can employ direct expressions for the pairwise error probability instead of using the Bhatacharyya bound. This results in tighter bounds on the error probability. The probability of selecting a path of weight d, when d is odd, over the all-zero path is the probability that the number of errors at these locations is greater than or equal to $(d+1)/2$. Therefore, the pairwise error probability is given by

$$P_2(d) = \sum_{k=(d+1)/2}^{d} \binom{d}{k} p^k (1-p)^{n-k} \tag{8.2–16}$$

If d is even, the incorrect path is selected when the number of errors exceeds $\frac{1}{2}d$. If the number of errors equals $\frac{1}{2}d$, there is a tie between the metrics in the two paths, which may be resolved by randomly selecting one of the paths; thus, an error occurs one-half the time. Consequently, the pairwise error probability in this case is given by

$$P_2(d) = \frac{1}{2}\binom{d}{\frac{1}{2}d} p^{d/2}(1-p)^{d/2} + \sum_{k=d/2+1}^{d} \binom{d}{k} p^k (1-p)^{n-k} \tag{8.2–17}$$

The error probability is bounded by

$$P_e \le \sum_{d=d_{\text{free}}}^{\infty} a_d P_2(d) \tag{8.2–18}$$

where $P_2(d)$ is substituted from Equations 8.2–16 and 8.2–17, for odd and even values of d, respectively.

A similar tighter bound for the bit error probability can also be derived by using the same approach. The result is given by

$$P_b \le \frac{1}{k} \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \tag{8.2–19}$$

where β_d are coefficients of Z^d in the expansion of $\frac{\partial}{\partial Y}T(Y, Z)$ computed at $Y = 1$.

A comparison of the error probability for the rate $1/3$, $K = 3$ convolutional code with soft decision decoding and hard decision decoding is made in Figure 8.2–1. Note that the upper bound given by Equation 8.2–15 for hard decision decoding is less than 1 dB above the tighter upper bound given by Equation 8.2–19 in conjunction with Equations 8.2–16 and 8.2–17. The advantage of the Bhatacharyya bound is its

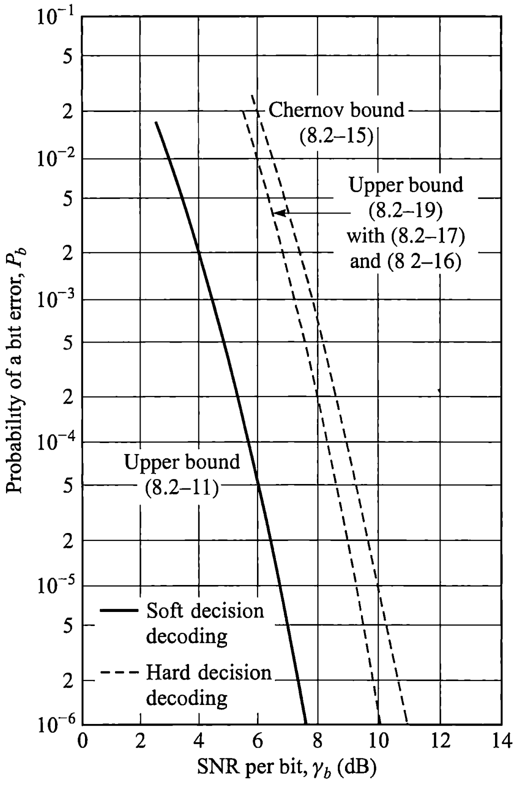


FIGURE 8.2–1
Comparison of soft decision and hard decision decoding for $K = 3$, $k = 1$, $n = 3$ convolutional code.

computational simplicity. In comparing the performance between soft decision and hard decision decoding, note that the difference obtained from the upper bounds is approximately 2.5 dB for $10^{-6} \leq P_b \leq 10^{-2}$.

Finally, we should mention that the ensemble average error rate performance of a convolutional code on a discrete memoryless channel, just as in the case of a block code, can be expressed in terms of the cutoff rate parameter R_0 as (for the derivation, see Viterbi and Omura (1979))

$$P_b \leq \frac{(q-1)\,q^{-KR_0/R_c}}{(1-q^{-(R_0-R_c)/R_c})^2}, \qquad R_c \leq R_0 \tag{8.2–20}$$

where q is the number of channel input symbols, K is the constraint length of the code, R_c is the code rate, and R_0 is the cutoff rate defined in Chapter 6. Therefore, conclusions reached by computing R_0 for various channel conditions apply to both block codes and convolutional codes.

8.3
DISTANCE PROPERTIES OF BINARY CONVOLUTIONAL CODES

In this subsection, we shall tabulate the minimum free distance and the generators for several binary, short-constraint-length convolutional codes for several code rates. These binary codes are optimal in the sense that, for a given rate and a given constraint length, they have the largest possible d_{free}. The generators and the corresponding values of d_{free} tabulated below have been obtained by Odenwalder (1970), Larsen (1973), Paaske (1974), and Daut et al. (1982) using computer search methods.

Heller (1968) has derived a relatively simple upper bound on the minimum free distance of a rate $1/n$ convolutional code. It is given by

$$d_{\text{free}} \leq \min_{l>1} \left\lfloor \frac{2^{l-1}}{2^l-1}(K+l-1)n \right\rfloor \tag{8.3–1}$$

where $\lfloor x \rfloor$ denotes the largest integer contained in x. For purposes of comparison, this upper bound is also given in the tables for the rate $1/n$ codes. For rate k/n convolutional codes, Daut et al. (1982) have given a modification to Heller's bound. The values obtained from this upper bound for k/n are also tabulated.

Tables 8.3–1 to 8.3–7 list the parameters of rate $1/n$ convolutional codes for $n = 2, 3, \ldots, 8$. Tables 8.3–8 to 8.3–11 list the parameters of several rate k/n convolutional codes for $k \leq 4$ and $n \leq 8$.

8.4
PUNCTURED CONVOLUTIONAL CODES

In some practical applications, there is a need to employ high-rate convolutional codes, e.g., rates of $(n-1)/n$. As we have observed, the trellis for such high-rate codes has 2^{n-1} branches that enter each state. Consequently, there are 2^{n-1} metric computations per state that must be performed in implementing the Viterbi algorithm and as many

TABLE 8.3–1
Rate 1/2 Maximum Free Distance Codes

Constraint Length K	Generators in Octal		d_{free}	Upper Bound on d_{free}
3	5	7	5	5
4	15	17	6	6
5	23	35	7	8
6	53	75	8	8
7	133	171	10	10
8	247	371	10	11
9	561	753	12	12
10	1,167	1,545	12	13
11	2,335	3,661	14	14
12	4,335	5,723	15	15
13	10,533	17,661	16	16
14	21,675	27,123	16	17

Sources: Odenwalder (1970) and Larsen (1973).

comparisons of the updated metrics to select the best path at each state. Therefore, the implementation of the decoder of a high-rate code can be very complex.

The computational complexity inherent in the implementation of the decoder of a high-rate convolutional code can be avoided by designing the high-rate code from a low-rate code in which some of the coded bits are deleted from transmission. The deletion of selected coded bits at the output of a convolutional encoder is called *puncturing*, as previously discussed in Section 7.8–2. Thus, one can generate high-rate convolutional codes by puncturing rate $1/n$ codes with the result that the decoder maintains the low complexity of the rate $1/n$ code. We note, of course, that puncturing a code reduces the free distance of the rate $1/n$ code by some amount that depends on the degree of puncturing.

The puncturing process may be described as periodically deleting selected bits from the output of the encoder, thus creating a periodically time-varying trellis code.

TABLE 8.3–2
Rate 1/3 Maximum Free Distance Codes

Constraint Length K	Generators in Octal			d_{free}	Upper Bound on d_{free}
3	5	7	7	8	8
4	13	15	17	10	10
5	25	33	37	12	12
6	47	53	75	13	13
7	133	145	175	15	15
8	225	331	367	16	16
9	557	663	711	18	18
10	1,117	1,365	1,633	20	20
11	2,353	2,671	3,175	22	22
12	4,767	5,723	6,265	24	24
13	10,533	10,675	17,661	24	24
14	21,645	35,661	37,133	26	26

Sources: Odenwalder (1970) and Larsen (1973).

TABLE 8.3–3
Rate 1/4 Maximum Free Distance Codes

Constraint Length K	Generators in Octal				d_{free}	Upper Bound on d_{free}
3	5	7	7	7	10	10
4	13	15	15	17	13	15
5	25	27	33	37	16	16
6	53	67	71	75	18	18
7	135	135	147	163	20	20
8	235	275	313	357	22	22
9	463	535	733	745	24	24
10	1,117	1,365	1,633	1,653	27	27
11	2,327	2,353	2,671	3,175	29	29
12	4,767	5,723	6,265	7,455	32	32
13	11,145	12,477	15,537	16,727	33	33
14	21,113	23,175	35,527	35,537	36	36

Source: Larsen (1973).

TABLE 8.3–4
Rate 1/5 Maximum Free Distance Codes

Constraint Length K	Generators in Octal					d_{free}	Upper Bound on d_{free}
3	7	7	7	5	5	13	13
4	17	17	13	15	15	16	16
5	37	27	33	25	35	20	20
6	75	71	73	65	57	22	22
7	175	131	135	135	147	25	25
8	257	233	323	271	357	28	28

Source: Daut et al. (1982).

TABLE 8.3–5
Rate 1/6 Maximum Free Distance Codes

Constraint Length K	Generators in Octal			d_{free}	Upper Bound on d_{free}
3	7	7	7	16	16
	7	5	5		
4	17	17	13	20	20
	13	15	15		
5	37	35	27	24	24
	33	25	35		
6	73	75	55	27	27
	65	47	57		
7	173	151	135	30	30
	135	163	137		
8	253	375	331	34	34
	235	313	357		

Source: Daut et al. (1982).

TABLE 8.3–6
Rate 1/7 Maximum Free Distance Codes

Constraint Length K	Generators in Octal				d_{free}	Upper Bound on d_{free}
3	7	7	7	7	18	18
	5	5	5			
4	17	17	13	13	23	23
	13	15	15			
5	35	27	25	27	28	28
	33	35	37			
6	53	75	65	75	32	32
	47	67	57			
7	165	145	173	135	36	36
	135	147	137			
8	275	253	375	331	40	40
	235	313	357			

Source: Daut et al. (1982).

TABLE 8.3–7
Rate 1/8 Maximum Free Distance Codes

Constraint Length K	Generators in Octal				d_{free}	Upper Bound on d_{free}
3	7	7	5	5	21	21
	5	7	7	7		
4	17	17	13	13	26	26
	13	15	15	17		
5	37	33	25	25	32	32
	35	33	27	37		
6	57	73	51	65	36	36
	75	47	67	57		
7	153	111	165	173	40	40
	135	135	147	137		
8	275	275	253	371	45	45
	331	235	313	357		

Source: Daut et al. (1982).

TABLE 8.3–8
Rate 2/3 Maximum Free Distance Codes

Constraint Length K	Generators in Octal			d_{free}	Upper Bound on d_{free}
2	17	6	15	3	4
3	27	75	72	5	6
4	236	155	337	7	7

Source: Daut et al. (1982).

TABLE 8.3–9
Rate $k/5$ Maximum Free Distance Codes

Rate	Constraint Length K	Generators in Octal					d_{free}	Upper Bound on d_{free}
2/5	2	17	07	11	12	04	6	6
	3	27	71	52	65	57	10	10
	4	247	366	171	266	373	12	12
3/5	2	35	23	75	61	47	5	5
4/5	2	237	274	156	255	337	3	4

Source: Daut et al. (1982).

TABLE 8.3–10
Rate $k/7$ Maximum Free Distance Codes

Rate	Constraint Length K	Generators in Octal				d_{free}	Upper Bound on d_{free}
2/7	2	05	06	12	15	9	9
		15	13	17			
	3	33	55	72	47	14	14
		25	53	75			
	4	312	125	247	366	18	18
		171	266	373			
3/7	2	45	21	36	62	8	8
		57	43	71			
4/7	2	130	067	237	274	6	7
		156	255	337			

Source: Daut et al. (1982).

TABLE 8.3–11
Rate 3/4 and 3/8 Maximum Free Distance Codes

Rate	Constraint Length K	Generators in Octal				d_{free}	Upper Bound on d_{free}
3/4	2	13	25	61	47	4	4
3/8	2	15	42	23	61	8	8
		51	36	75	47		

Source: Daut et al. (1982).

We begin with a rate $1/n$ parent code and define a *puncturing period* P, corresponding to P input information bits to the encoder. Hence, in one period, the encoder outputs nP coded bits. Associated with the nP encoded bits is a *puncturing matrix* $\boldsymbol{P}$ of the form

$$\boldsymbol{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1P} \\ p_{21} & p_{22} & \cdots & p_{2P} \\ \vdots & \vdots & \vdots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nP} \end{bmatrix} \tag{8.4–1}$$

where each column of $\boldsymbol{P}$ corresponds to the n possible output bits from the encoder for each input bit and each element of $\boldsymbol{P}$ is either 0 or 1. When $p_{ij} = 1$, the corresponding output bit from the encoder is transmitted. When $p_{ij} = 0$, the corresponding output bit from the encoder is deleted. Thus, the code rate is determined by the period P and the number of bits deleted.

If we delete N bits out of nP, the code rate is $P/(nP - N)$, where N may take any integer value in the range 0 to $(n-1)P - 1$. Hence, the achievable code rates are

$$R_c = \frac{P}{P+M}, \qquad M = 1, 2, \ldots, (n-1)P \tag{8.4–2}$$

EXAMPLE 8.4–1. Let us construct a rate $\frac{3}{4}$ code by puncturing the output of the rate $\frac{1}{3}$, $K = 3$ encoder shown in Figure 8.1–2. There are many choices for P and M in Equation 8.4–2 to achieve the desired rate. We may take the smallest value of P, namely, $P = 3$. Then out of every $nP = 9$ output bits, we delete $N = 5$ bits. Thus, we achieve a rate $\frac{3}{4}$ punctured convolutional code. As the puncturing matrix, we may select P as

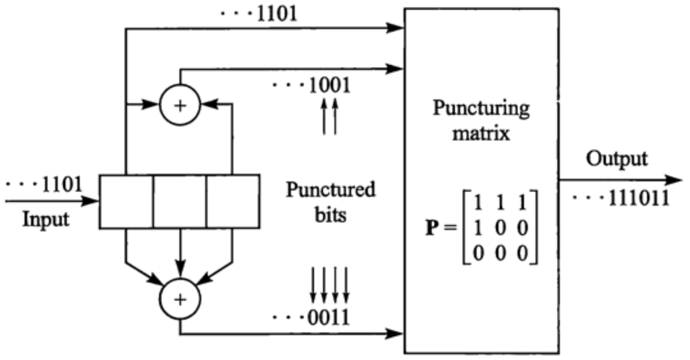
$$\boldsymbol{P} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{8.4–3}$$

Figure 8.4–1 illustrates the generation of the punctured code from the rate $\frac{1}{3}$ parent code. The corresponding trellis for the punctured code is also shown in Figure 8.4–1.

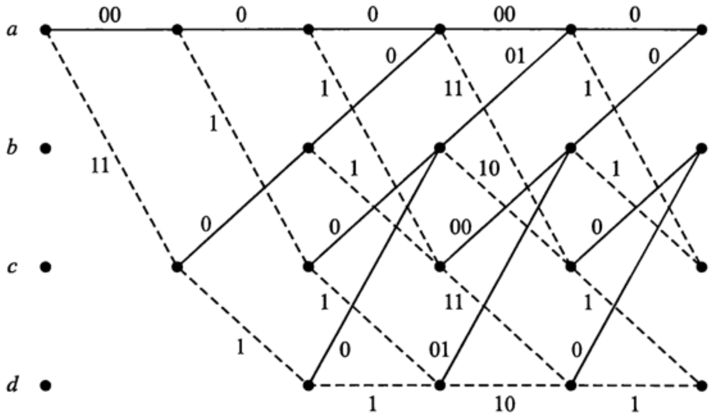
In the example given above, the puncturing matrix was selected arbitrarily. However, some puncturing matrices are better than others in that the trellis paths have better Hamming distance properties. A computer search is usually employed to find good puncturing matrices. Generally, the high-rate punctured convolutional codes generated in this manner have a free distance that is either equal to or 1 bit less than the best same high-rate convolutional code obtained directly without puncturing.

Yasuda et al. (1984), Hole (1988), Lee (1988), Haccoun and Bégin (1989), and Bégin et al. (1990) have investigated the construction and properties of small and large constraint length punctured convolutional codes generated from low-rate codes. In general, high-rate codes with good distance properties are obtained by puncturing rate $\frac{1}{2}$ maximum free distance codes. For example, in Table 8.4–1 we list the puncturing matrices for code rates of $\frac{2}{3} \le R_c \le \frac{7}{8}$ which are obtained by puncturing rate $\frac{1}{2}$ codes with constraint lengths $3 \le K \le 9$. The free distances of the punctured codes are also given in the table. Punctured convolutional codes for additional rates and larger constraint lengths may be found in the papers referred to above.

The decoding of punctured convolutional codes is performed in the same manner as the decoding of the low-rate $1/n$ parent code, using the trellis of the $1/n$ code. The path metrics in the trellis for soft decision decoding are computed in the conventional way as described previously. When one or more bits in a branch are punctured, the corresponding branch metric increment is computed based on the nonpunctured bits; thus, the punctured bits do not contribute to the branch metrics. Error events in a punctured code are generally longer than error events in the low-rate $1/n$ parent code. Consequently, the decoder must wait longer than five constraint lengths before making


· · · 1101
· · · 1001
Puncturing matrix
Punctured bits
Output
· · · 111011
· · · 1101
Input
· · · 0011
$\mathbf{P} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

(a) Encoder


a
b
c
d
00
0
0
00
0
11
1
1
0
01
0
11
1
1
0
0
1
10
1
00
0
0
1
11
1
1
0
01
0
1
10
1

(b) Trellis of punctured code

FIGURE 8.4–1
Generation of a rate 3/4 punctured code from a rate 1/3 convolutional code.

TABLE 8.4–1
Puncturing Matrices for Code Rates of $2/3 \le R_c \le 7/8$ from Rate 1/2 Code

	Rate 2/3		Rate 3/4		Rate 4/5		Rate 5/6		Rate 6/7		Rate 7/8	
K	***P***	d_{free}	***P***	d_{free}	***P***	d_{free}	***P***	d_{free}	***P***	d_{free}	***P***	d_{free}
3	10	3	101	3	1011	2	10111	2	101111	2	1011111	2
	11		110		1100		11000		110000		1100000	
4	11	4	110	4	1011	3	10100	3	100011	2	1000010	2
	10		101		1100		11011		111100		1111101	
5	11	4	101	3	1010	3	10111	3	101010	3	1010011	3
	10		110		1101		11000		110101		1101100	
6	10	6	100	4	1000	4	10000	4	110110	3	1011101	3
	11		111		1111		11111		101001		1100010	
7	11	6	110	5	1111	4	11011	4	111010	3	1111010	3
	10		101		1000		10101		100101		1000101	
8	10	7	110	6	1010	5	11100	4	101001	4	1010100	4
	11		101		1101		10011		110110		1101011	
9	11	7	111	6	1101	5	10110	5	110110	4	1101011	4
	10		100		1010		11001		101001		1010100	

final decisions on the received bits. For soft decision decoding, the performance of the punctured codes is given by the error probability (upper bound) expression in Equation 8.2–15 for the bit error probability.

An approach for the design of good punctured codes is to search and select puncturing matrices that yield the maximum free distance. A somewhat better approach is to determine the weight spectrum $\{\beta_d\}$ of the dominant terms of the punctured code and to calculate the corresponding bit error probability bound. The code corresponding to the puncturing matrix that results in the best error rate performance may then be selected as the best punctured code, provided that it is not catastrophic. In general, in determining the weight spectrum for a punctured code, it is necessary to search through a larger number of paths over longer lengths than the underlying low-rate $1/n$ parent code. Weight spectra for several punctured codes are given in the papers by Haccoun and Bégin (1989) and Bégin et al. (1990).

8.4–1 Rate-Compatible Punctured Convolutional Codes

In the transmission of compressed digital speech signals and in some other applications, there is a need to transmit some groups of information bits with more redundancy than others. In other words, the different groups of information bits require unequal error protection to be provided in the transmission of the information sequence, where the more important bits are transmitted with more redundancy. Instead of using separate codes to encode the different groups of bits, it is desirable to use a single code that has variable redundancy. This can be accomplished by puncturing the same low-rate $1/n$ convolutional code by different amounts as described by Hagenauer (1988). The puncturing matrices are selected to satisfy a rate compatibility criterion, where the basic requirement is that lower-rate codes (higher redundancy) transmit the same coded bits as all higher-rate codes plus additional bits. The resulting codes obtained from a single rate $1/n$ convolutional code are called *rate-compatible punctured convolutional* (RCPC) *codes*.

EXAMPLE 8.4–2. From the rate $\frac{1}{3}$, $K = 4$ maximum free distance convolutional code, let us construct an RCPC code. The RCPC codes for this example are taken from the paper of Hagenauer (1988), who selected $P = 8$ and generated codes of rates ranging from $\frac{4}{11}$ to $\frac{8}{9}$. The puncturing matrices are listed in Table 8.4–2. Note that the rate $\frac{1}{2}$ code has a puncturing matrix with all zeros in the third row. Hence all bits from the third branch of the rate $\frac{1}{3}$ encoder are deleted. Higher code rates are obtained by deleting additional bits from the second branch of the rate $\frac{1}{3}$ encoder. However, note that when a 1 appears in a puncturing matrix of a high-rate code, a 1 also appears in the same position for all lower-rate codes.

In applying RCPC codes to systems that require unequal error protection of the information sequence, we may format the groups of bits into a frame structure, as suggested by Hagenauer et al. (1990) and illustrated in Figure 8.4–2, where, for example, three groups of bits of different lengths N_1, N_2, and N_3 are arranged in order of their corresponding specified error protection probabilities $p_1 > p_2 > p_3$. Each frame is terminated after the last group of information bits (N_3) by $K - 1$ zeros, which result

TABLE 8.4–2
Rate-Compatible Punctured Convolutional Codes Constructed from Rate 1/3, $K = 4$ Code with $P = 8$ $R_c = P/(P + M)$, $M = 1, 2, 4, 6, 8, 10, 12, 14$

Rate	Puncturing Matrix P
$\frac{1}{3}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \end{bmatrix}$
$\frac{4}{11}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \\ 1&1&1&0&1&1&1&0 \end{bmatrix}$
$\frac{2}{5}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \\ 1&0&1&0&1&0&1&0 \end{bmatrix}$
$\frac{4}{9}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \\ 1&0&0&0&1&0&0&0 \end{bmatrix}$
$\frac{1}{2}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&1&1&1&1&1 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$
$\frac{4}{7}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&1&1&0&1&1&1&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$
$\frac{4}{6}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&0&1&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$
$\frac{4}{5}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$
$\frac{8}{9}$	$\begin{bmatrix} 1&1&1&1&1&1&1&1 \\ 1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \end{bmatrix}$

in overhead bits that are used for the purpose of terminating the trellis in the all-zero state. We then select an appropriate set of RCPC codes that satisfy the error protection requirements, i.e., the specified error probabilities $\{p_k\}$. In our example, the group of bits will be encoded by the use of three puncturing matrices having period P corresponding to a set of RCPC codes generated from a rate $1/n$ code. Thus, the bits requiring the least

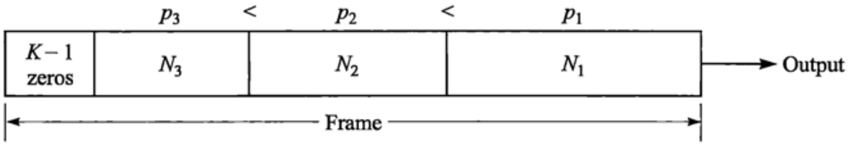


FIGURE 8.4–2
Frame structure for transmitting data with unequal error protection.

protection are transmitted first, followed by the bits requiring the next-higher level of protection, up to the group of bits requiring the highest level of protection, followed by the all-zero terminating sequence. All rate transitions occur within the frame without compromising the designed error rate performance requirements. As in the encoding, the bits within a frame are decoded by a single Viterbi algorithm using the trellis of the rate $1/n$ code and performing metric computations based on the appropriate puncturing matrix for each group of bits.

It can be shown (see Problem 8.21) that the average effective code rate of this scheme is

$$R_{\mathrm{av}} = \frac{\sum_{j=1}^{J} N_j P}{\sum_{j=1}^{J} N_j(P + M_j) + (K-1)(P + M_J)} \tag{8.4–4}$$

where J is the number of groups of bits in the frame, P is the period of the RCPC codes, and the second term in the denominator corresponds to the overhead code bits which are transmitted with the lowest code rate (highest redundancy).

8.5
OTHER DECODING ALGORITHMS FOR CONVOLUTIONAL CODES

The Viterbi algorithm described in Section 8.2–1 is the optimum decoding algorithm (in the sense of maximum-likelihood decoding of the entire sequence) for convolutional codes. However, it requires the computation of 2^{kK} metrics at each node of the trellis and the storage of $2^{k(K-1)}$ metrics and $2^{k(K-1)}$ surviving sequences, each of which may be about $5kK$ bits long. The computational burden and the storage required to implement the Viterbi algorithm make it impractical for convolutional codes with large constraint length.

Prior to the discovery of the optimum algorithm by Viterbi, a number of other algorithms had been proposed for decoding convolutional codes. The earliest was the sequential decoding algorithm originally proposed by Wozencraft (1957), further treated by Wozencraft and Reiffen (1961), and subsequently modified by Fano (1963).

Sequential decoding algorithm The Fano sequential decoding algorithm searches for the most probable path through the tree or trellis by examining one path at a time. The increment added to the metric along each branch is proportional to the probability of the received signal for that branch, just as in Viterbi decoding, with the exception that an additional negative constant is added to each branch metric. The value of this constant is selected such that the metric for the correct path will increase on the average, while the metric for any incorrect path will decrease on the average. By comparing the metric of a candidate path with a moving (increasing) threshold, Fano's algorithm detects and discards incorrect paths.

To be more specific, let us consider a memoryless channel. The metric for the ith path through the tree or trellis from the first branch to branch B may be expressed as

$$CM^{(i)} = \sum_{j=1}^{B} \sum_{m=1}^{n} \mu_{jm}^{(i)} \tag{8.5–1}$$

where

$$\mu_{jm}^{(i)} = \log_2 \frac{p\left(r_{jm}|c_{jm}^{(i)}\right)}{p(r_{jm})} - \mathcal{K} \tag{8.5–2}$$

In Equation 8.5–2, r_{jm} is the demodulator output sequence, $p(r_{jm}|c_{jm}^{(i)})$ denotes the PDF of r_{jm} conditional on the code bit $c_{jm}^{(i)}$ for the mth bit of the jth branch of the ith path, and $\mathcal{K}$ is a positive constant. $\mathcal{K}$ is selected as indicated above so that the incorrect paths will have a decreasing metric while the correct path will have an increasing metric on the average. Note that the term $p(r_{jm})$ in the denominator is independent of the code sequence, and, hence, may be subsumed in the constant factor.

The metric given by Equation 8.5–2 is generally applicable for either hard- or soft-decision decoding. However, it can be considerably simplified when hard-decision decoding is employed. Specifically, if we have a BSC with transition (error) probability p, the metric for each received bit, consistent with the form in Equation 8.5–2 is given by

$$\mu_{jm}^{(i)} = \begin{cases} \log_2[2(1-p)] - R_c & \left(\text{if } \tilde{r}_{jm} = c_{jm}^{(i)}\right) \\ \log_2 2p - R_c & \left(\text{if } \tilde{r} \neq c_{jm}^{(i)}\right) \end{cases} \tag{8.5–3}$$

where $\tilde{r}_{jm}$ is the hard-decision output from the demodulator, $c_{jm}^{(i)}$ is the mth code bit in the jth branch of the ith path in the tree, and R_c is the code rate. Note that this metric requires some (approximate) knowledge of the error probability.

EXAMPLE 8.5–1. Suppose we have a rate $R_c = 1/3$ binary convolutional code for transmitting information over a BSC with $p = 0.1$. By evaluating Equation 8.5–3 we find that

$$\mu_{jm}^{(i)} = \begin{cases} 0.52 & \left(\text{if } \tilde{r}_{jm} = c_{jm}^{(i)}\right) \\ -2.65 & \left(\text{if } \tilde{r}_{jm} \neq c_{jm}^{(i)}\right) \end{cases} \tag{8.5–4}$$

To simplify the computations, the metric in Equation 8.5–4 may be normalized. It is well approximated as

$$\mu_{jm}^{(i)} = \begin{cases} 1 & \left(\text{if } \tilde{r}_{jm} = c_{jm}^{(i)}\right) \\ -5 & \left(\text{if } \tilde{r}_{jm} \neq c_{jm}^{(i)}\right) \end{cases} \tag{8.5–5}$$

Since the code rate is 1/3, there are three output bits from the encoder for each input bit. Hence, the branch metric consistent with Equation 8.5–5 is

$$\mu_j^{(i)} = 3 - 6d$$

or, equivalently,

$$\mu_j^{(i)} = 1 - 2d \tag{8.5–6}$$

where d is the Hamming distance of the three received bits from the three branch bits. Thus, the metric $\mu_j^{(i)}$ is simply related to the Hamming distance between received bits and the code bits in the jth branch of the ith path.

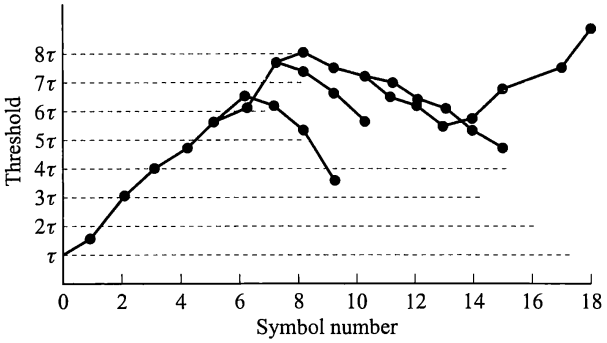


FIGURE 8.5–1
An example of the path search in sequential decoding. [*From Jordan (1996), © 1966 IEEE.*]

Initially, the decoder may be forced to start on the correct path by the transmission of a few known bits of data. Then it proceeds forward from node to node, taking the most probable (largest metric) branch at each node and increasing the threshold such that the threshold is never more than some preselected value, say τ, below the metric. Now suppose that the additive noise (for soft-decision decoding) or demodulation errors resulting from noise on the channel (for hard-decision decoding) cause the decoder to take an incorrect path because it appears more probable than the correct path. This is illustrated in Figure 8.5–1. Since the metrics of an incorrect path decrease on the average, the metric will fall below the current threshold, say τ_0. When this occurs, the decoder backs up and takes alternative paths through the tree or trellis, in order of decreasing branch metrics, in an attempt to find another path that exceeds the threshold τ_0. If it is successful in finding an alternative path, it continues along that path, always selecting the most probable branch at each node. On the other hand, if no path exists that exceeds the threshold τ_0, the threshold is reduced by an amount τ and the original path is retraced. If the original path does not stay above the new threshold, the decoder resumes its backward search for other paths. This procedure is repeated, with the threshold reduced by τ for each repetition, until the decoder finds a path that remains above the adjusted threshold. A simplified flow diagram of Fano's algorithm is shown in Figure 8.5–2.

The sequential decoding algorithm requires a buffer memory in the decoder to store incoming demodulated data during periods when the decoder is searching for alternate paths. When a search terminates, the decoder must be capable of processing demodulated bits sufficiently fast to empty the buffer prior to commencing a new search. Occasionally, during extremely long searches, the buffer may overflow. This causes loss of data, a condition that can be remedied by retransmission of the lost information. In this regard, we should mention that the cutoff rate R_0 has special meaning in sequential decoding. It is the rate above which the average number of decoding operations per decoded digit becomes infinite, and it is termed the *computational cutoff rate* R_{comp}. In practice, sequential decoders usually operate at rates near R_0.

The Fano sequential decoding algorithm has been successfully implemented in several communication systems. Its error rate performance is comparable to that of Viterbi decoding. However, in comparison with Viterbi decoding, sequential decoding has a significantly larger decoding delay. On the positive side, sequential decoding requires less storage than Viterbi decoding and, hence, it appears attractive for convolutional codes with a large constraint length. The issues of computational complexity and storage requirements for sequential decoding are interesting and have been thoroughly investigated. For an analysis of these topics and other characteristics of the Fano

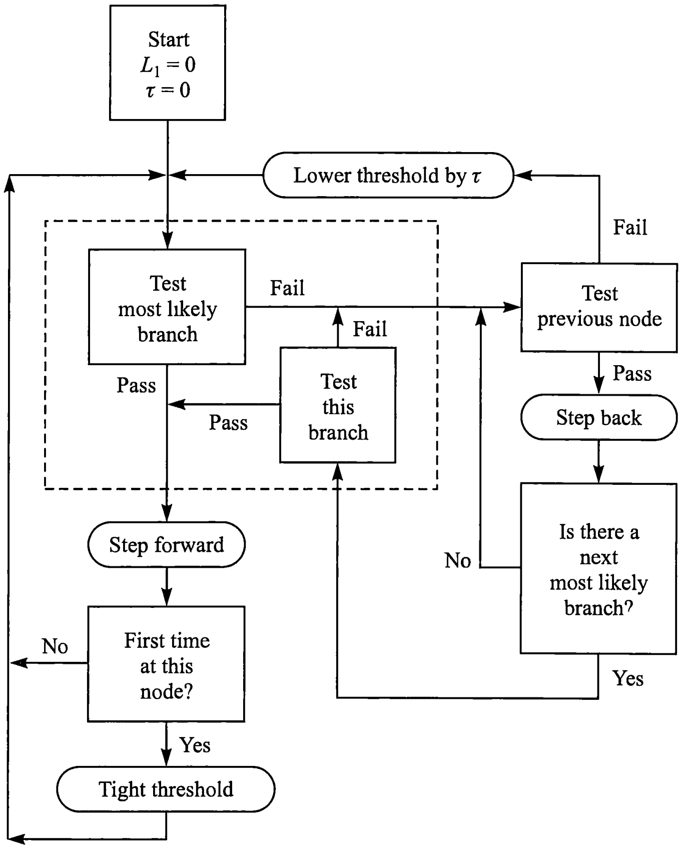

Start
$L_1 = 0$
$\tau = 0$
Lower threshold by τ
Fail
Test most likely branch
Fail
Fail
Test previous node
Pass
Pass
Test this branch
Pass
Step back
Step forward
No
Is there a next most likely branch?
No
First time at this node?
Yes
Yes
Tight threshold

FIGURE 8.5–2

A simplified flow diagram of Fano's algorithm. [*From Jordan (1966), © 1966 IEEE.*]

algorithm, the interested reader may refer to Gallager (1968), Wozencraft and Jacobs (1965), Savage (1966), and Forney (1974).

Stack algorithm Another type of sequential decoding algorithm, called a *stack algorithm*, has been proposed independently by Jelinek (1969) and Zigangirov (1966). In contrast to the Viterbi algorithm, which keeps track of $2^{(K-1)k}$ paths and corresponding metrics, the stack sequential decoding algorithm deals with fewer paths and their corresponding metrics. In a stack algorithm, the more probable paths are ordered according to their metrics, with the path at the top of the stack having the largest metric. At each step of the algorithm, only the path at the top of the stack is extended by one branch. This yields 2^k successors and their corresponding metrics. These 2^k successors along with the other paths are then reordered according to the values of the metrics, and all paths with metrics that fall below some preselected amount from the metric of the top path may be discarded. Then the process of extending the path with the largest metric is repeated. Figure 8.5–3 illustrates the first few steps in a stack algorithm.

It is apparent that when none of the 2^k extensions of the path with the largest metric remains at the top of the stack, the next step in the search involves the extension of another path that has climbed to the top of the stack. It follows that the algorithm does not necessarily advance by one branch through the trellis in every iteration. Consequently,

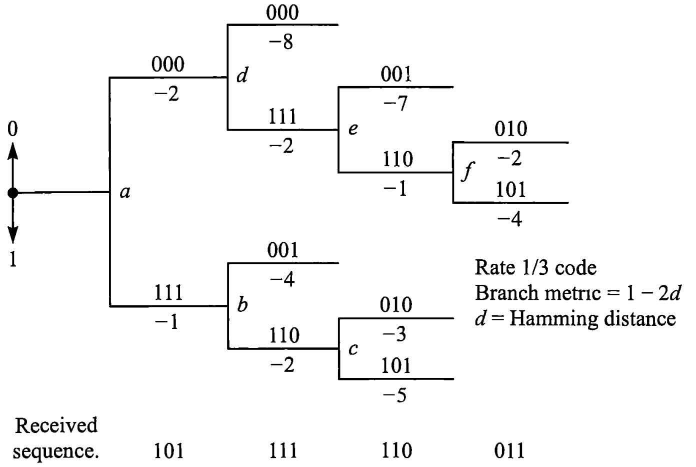


FIGURE 8.5–3
An example of the stack algorithm for decoding a rate 1/3 convolutional code.

Stack with accumulated path metrics

Step *a*	Step *b*	Step *c*	Step *d*	Step *e*	Step *f*
−1	−2	−3	−2	−1	−2
−3	−3	−3	−3	−3	−3
	−4	−4	−4	−4	−4
		−5	−5	−5	−4
			−8	−7	−5
				−8	−7
					−8

some amount of storage must be provided for newly received signals and previously received signals in order to allow the algorithm to extend the search along one of the shorter paths, when such a path reaches the top of the stack.

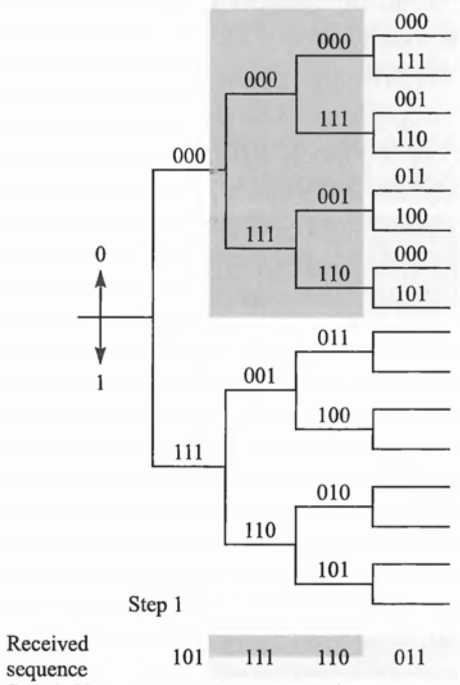
In a comparison of the stack algorithm with the Viterbi algorithm, the stack algorithm requires fewer metric computations, but this computational saving is offset to a large extent by the computations involved in reordering the stack after every iteration. In comparison with the Fano algorithm, the stack algorithm is computationally simpler, since there is no retracing over the same path as is done in the Fano algorithm. On the other hand, the stack algorithm requires more storage than the Fano algorithm.

Feedback decoding A third alternative to the optimum Viterbi decoder is a method called *feedback decoding* (Heller, 1975), which has been applied to decoding for a BSC (hard-decision decoding). In feedback decoding, the decoder makes a hard decision on the information bit at stage j based on metrics computed from stage j to stage $j + m$, where m is a preselected positive integer. Thus, the decision on the information bit is either 0 or 1 depending on whether the minimum Hamming distance path that begins at stage j and ends at stage $j + m$ contains a 0 or 1 in the branch emanating from stage j. Once a decision is made on the information bit at stage j, only that part of the tree that stems from the bit selected at stage j is kept (half the paths emanating from node j) and the remaining part is discarded. This is the feedback feature of the decoder.

The next step is to extend the part of the tree that has survived to stage $j+1+m$ and consider the paths from stage $j+1$ to $j+1+m$ in deciding on the bit at stage $j+1$. Thus, this procedure is repeated at every stage. The parameter m is simply the number of stages in the tree that the decoder looks ahead before making a hard decision. Since a large value of m results in a large amount of storage, it is desirable to select m as small as possible. On the other hand, m must be sufficiently large to avoid a severe degradation in performance. To balance these two conflicting requirements, m is usually selected in the range $K \leq m \leq 2K$, where K is the constraint length. Note that this decoding delay is significantly smaller than the decoding delay in a Viterbi decoder, which is usually about $5K$.

EXAMPLE 8.5–2. Let us consider the use of a feedback decoder for the rate 1/3 convolutional code shown in Figure 8.1–2. Figure 8.5–4 illustrates the tree diagram and the operation of the feedback decoder for $m = 2$. That is, in decoding the bit at branch j, the decoder considers the paths at branches j, $j+1$, and $j+2$. Beginning with the first branch, the decoder computes eight metrics (Hamming distances) and decides that the bit for the first branch is 0 if the minimum distance path is contained in the upper part of the tree, and 1 if the minimum distance path is contained in the lower part of the tree. In this example, the received sequence for the first three branches is assumed to be 101111110, so that the minimum distance path is in the upper part of the tree. Hence, the first output bit is 0.

The next step is to extend the upper part of the tree (the part of the tree that has survived) by one branch, and to compute the eight metrics for branches 2, 3, and 4. For the assumed received sequence 111110011, the minimum-distance path is contained in the lower part of the section of the tree that survived from the first step. Hence, the second output bit is 1. The third step is to extend this lower part of the tree and to repeat the procedure described for the first two steps.



Step 1: Upper-tree metrics: 7,6,5,2*; lower-tree metrics: 5,4,3,4 → 0
Step 2: Upper-tree metrics: 7,6,5,6; lower-tree metrics: 3,6,1*,2 → 1

FIGURE 8.5–4
An example of feedback decoding for a rate 1/3 convolutional code.

Instead of computing metrics as described above, a feedback decoder for the BSC may be efficiently implemented by computing the syndrome from the received sequence and using a table lookup method for correcting errors. This method is similar to the one described for decoding block codes. For some convolutional codes, the feedback decoder simplifies to a form called a *majority logic decoder* or a *threshold decoder* (Massey (1963); Heller (1975)).

Soft-output algorithms The outputs of the Viterbi algorithm and the three algorithms described in this section are hard decisions. In some cases, it is desirable to have soft outputs from the decoder. This is the case if the decoding is being performed on an inner code in a concatenated code, where it is desirable to provide soft decisions to the input of the outer decoder. This is also the case in iterative decoding of concatenated codes, previously discussed in the context of block codes in Section 7.13–2, and further treated in the context of convolutional codes in Section 8.9–2.

The optimum metric that provides a measure of the reliability of symbol decisions is the a posteriori probability of the detected symbol conditioned on the received signal vector $\boldsymbol{r} = \{r_{jm}, m = 1, 2, \cdots, n; j = 1, 2, \cdots B\}$, where $\{r_{jm}\}$ is the sequence of soft outputs from the demodulator, n is the number of output symbols from the encoder for each k input symbols, and j is the branch index. For example, the output of the demodulator for a binary convolutional code and binary PSK modulation in an AWGN channel is

$$r_{jm} = (2c_{jm} - 1)\sqrt{\mathcal{E}_c} + n_{jm} \tag{8.5–7}$$

where $\{c_{jm} = 0, 1\}$ are the output bits from the encoder. Given the received vector $\mathbf{r}$, decisions on the transmitted information bits are based on the maximum a posteriori probability (MAP), which may be expressed as

$$P(x_i = 0|\boldsymbol{r}) = 1 - P(x_i = 1|\boldsymbol{r}) \tag{8.5–8}$$

where x_i denotes the ith information bit in the sequence. Thus, under the MAP criterion, a decision is made on a symbol-by-symbol basis by selecting the information symbol, or bit in this case, corresponding to the largest a posteriori probability. If the a posteriori probabilities for the possible transmitted symbols are nearly the same, the decision is unreliable. Hence, the a posteriori probability associated with the decided symbol (the hard decision) is the soft output from the decoder that provides a measure, or metric, for the reliability of the hard decision. Since the MAP criterion minimizes the probability of a symbol error, the a posteriori probability metric is the optimum soft output of the decoder.

An algorithm for recursively computing the a posteriori probabilities for each received symbol given the received signal sequence $\boldsymbol{r}$ from the demodulator has been described in the paper by Bahl, Cocke, Jelinek, and Raviv (1974). This symbol-by-symbol decoding algorithm, called the BCJR algorithm, is based on the MAP criterion and provides a hard decision on each received symbol and the a posteriori probability metric that serves as a measure for the reliability of the hard decision. The BCJR algorithm is described in Section 8.8.

In contrast to the MAP symbol-by-symbol detection criterion, the Viterbi algorithm selects the sequence that maximizes the probability $p(\boldsymbol{r}|\boldsymbol{x})$, where $\boldsymbol{x}$ is the vector of information bits. In this case, the soft output metric is the Euclidean distance associated with the sequence of received symbols, as opposed to the individual symbols. However, it is possible to derive symbol metrics from the sequence or path metrics. Hagenauer and Hoeher (1989) devised a soft-output Viterbi algorithm (SOVA) that provides a reliability metric for each decoded symbol. The SOVA is based on the observation that the probability that a hard decision on a given symbol at the output of the Viterbi algorithm is correct is proportional to the difference in path metrics between a surviving sequence and its associated nonsurviving sequences. This observation allows us to form an estimate of the error probability, or the probability of a correct decision, for each symbol by comparing the path metrics of the surviving path with the path metrics of nonsurviving paths.

For example, let us consider a binary convolutional code with binary PSK modulation. Since the Viterbi algorithm makes decisions with a decoding delay δ, at time $t = i + \delta$ the Viterbi decoder outputs the bit $\hat{x}_{is}$ from the most probable surviving sequence. When we trace back along the surviving path from t to $t - \delta$, we observe that we have discarded $\delta + 1$ paths. Let us consider the jth discarded path and its corresponding bit x_{ij} at time $t = i$. If $\hat{x}_{is} \neq x_{ij}$, let ψ_j ($\psi_j \geq 0$) be equal to the difference in the path metrics between the surviving path and the jth discarded path. If $\hat{x}_{is} = x_{ij}$, let $\psi_j = \infty$. This comparison is performed for all discarded paths. From the set $\{\psi_j,\ j = 0, 1, 2, \cdots, \delta\}$ we select the smallest value, defined as $\psi_{\min} = \min\{\psi_0, \psi_1, \cdots, \psi_\delta\}$. Then, the probability of error for the bit $\hat{x}_{is}$ is approximated as

$$\hat{P}_e = \frac{1}{1 + e^{\psi_{\min}}} \tag{8.5–9}$$

Note that if $\psi_{\min}$ is very small, $\hat{P}_e \approx \frac{1}{2}$, so the decision on $\hat{x}_{is}$ is unreliable. Thus, $\hat{P}_e$ provides a reliability metric for the hard decisions at the output of the Viterbi algorithm. We note, however, that $\hat{P}_e$ is only an approximation to the true error probability. That is, $\hat{P}_e$ is not the optimum soft-output metric for the hard decisions at the output of the Viterbi algorithm. In fact, it has been observed in a paper by Wang and Wicker (1996) that $\hat{P}_e$ underestimates the true error probability at low SNR. Nevertheless, this soft-output metric from the Viterbi algorithm leads to a significant improvement in the performance of the decoder in a concatenated code.

From Equation 8.5–9 we can obtain an estimate of the probability of a correct decision as

$$\hat{P}_c = 1 - \hat{P}_e = \frac{e^{\psi_{\min}}}{1 + e^{\psi_{\min}}} \tag{8.5–10}$$

8.6

PRACTICAL CONSIDERATIONS IN THE APPLICATION OF CONVOLUTIONAL CODES

Convolutional codes are widely used in many practical applications of communication system design. Viterbi decoding is predominantly used for short constraint lengths ($K \leq 10$), while sequential decoding is used for long-constraint-length codes, where

TABLE 8.6–1
Upper Bounds on Coding Gain for Soft-Decision Decoding of Some Convolutional Codes

Rate 1/2 codes			Rate 1/3 codes		
Constraint Length K	d_{free}	**Upper bound, dB**	**Constraint Length K**	d_{free}	**Upper bound, dB**
3	5	3.98	3	8	4.26
4	6	4.77	4	10	5.23
5	7	5.44	5	12	6.02
6	8	6.02	6	13	6.37
7	10	6.99	7	15	6.99
8	10	6.99	8	16	7.27
9	12	7.78	9	18	7.78
10	12	7.78	10	20	8.24

the complexity of Viterbi decoding becomes prohibitive. The choice of constraint length is dictated by the desired coding gain.

From the error probability results for soft-decision decoding given by Equations 8.2–11, 8.2–12, and 8.2–13, it is apparent that the coding gain achieved by a convolutional code over an uncoded binary PSK or QPSK system is

$$\text{Coding gain} \leq 10\log_{10}(R_c d_{\text{free}})$$

We also know that the minimum free distance d_{free} can be increased either by decreasing the code rate or by increasing the constraint length, or both. Table 8.6–1 provides a list of upper bounds on the coding gain for several convolutional codes. For purposes of comparison, Table 8.6–2 lists the actual coding gains for several short-constraint-length convolutional codes with Viterbi decoding. It should be noted that the coding gain increases toward the asymptotic limit as the SNR per bit increases.

These results are based on soft-decision Viterbi decoding. If hard-decision decoding is used, the coding gains are reduced by approximately 2 dB for the AWGN channel.

Larger coding gains than those listed in Tables 8.6–1 and 8.6–2 are achieved by employing long-constraint-length convolutional codes, e.g., $K = 50$, and decoding such codes by sequential decoding. Invariably, sequential decoders are implemented

TABLE 8.6–2
Coding Gain (dB) for Soft-Decision Viterbi Decoding

	$\mathcal{E}_b/N_0$ **Uncoded, dB**	$R_c = 1/3$		$R_c = 1/2$			$R_c = 2/3$		$R_c = 3/4$	
P_b		$K = 8$	$K = 8$	$K = 5$	$K = 6$	$K = 7$	$K = 6$	$K = 8$	$K = 6$	$K = 9$
10^{-3}	6.8	4.2	4.4	3.3	3.5	3.8	2.9	3.1	2.6	2.6
10^{-5}	9.6	5.7	5.9	4.3	4.6	5.1	4.2	4.6	3.6	4.2
10^{-7}	11.3	6.2	6.5	4.9	5.3	5.8	4.7	5.2	3.9	4.8

Source: Jacobs (1974); © IEEE

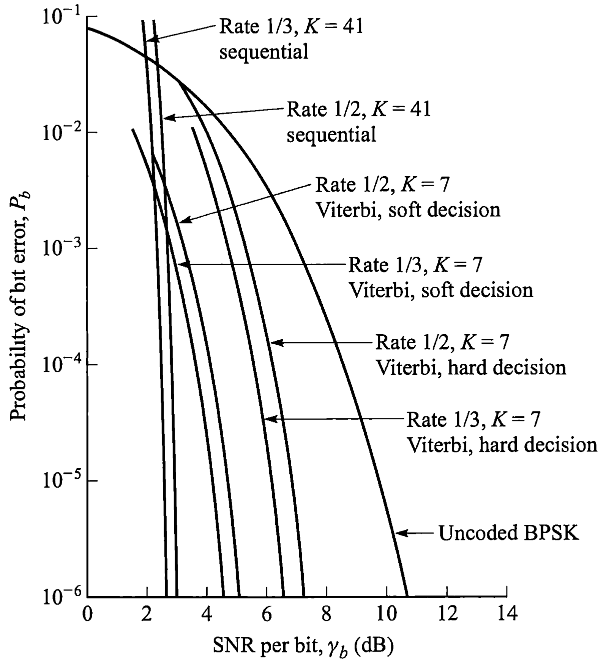


FIGURE 8.6–1
Performance of rate 1/2 and rate 1/3 Viterbi and sequential decoding. [*From Omura and Levitt (1982). © 1982 IEEE.*]

for hard-decision decoding to reduce complexity. Figure 8.6–1 illustrates the error rate performance of several constraint-length $K = 7$ convolutional codes for rates 1/2 and 1/3 and for sequential decoding (with hard decisions) of a rate 1/2 and a rate 1/3 constraint-length $K = 41$ convolutional codes. Note that the $K = 41$ codes achieve an error rate of 10^{-6} at 2.5 and 3 dB, which are within 4–4.5 dB of the channel capacity limit, i.e., in the vicinity of the cutoff rate limit. However, the rate 1/2 and rate 1/3, $K = 7$ codes with soft-decision Viterbi decoding operate at about 5 and 4.4 dB at 10^{-6}, respectively. These short-constraint-length codes achieve a coding gain of about 6 dB at 10^{-6}, while the long-constraint-length codes gain about 7.5–8 dB.

Two important issues in the implementation of Viterbi decoding are

1. The effect of path memory truncation, which is a desirable feature that ensures a fixed decoding delay.
2. The degree of quantization of the input signal to the Viterbi decoder.

As a rule of thumb, we stated that path memory truncation to about five constraint lengths has been found to result in negligible performance loss. Figure 8.6–2 illustrates the performance obtained by simulation for rate 1/2, constraint-lengths $K = 3$, 5, and 7 codes with memory path length of 32 bits. In addition to path memory truncation, the computations were performed with eight-level (three bits) quantized input signals from the demodulator. The broken curves are performance results obtained from the upper bound in the bit error rate given by Equation 8.2–12. Note that the simulation results are close to the theoretical upper bounds, which indicate that the degradation due to path memory truncation and quantization of the input signal has a minor effect on performance (0.20–0.30 dB).

Figure 8.6–3 illustrates the bit error rate performance obtained via simulation for hard-decision decoding of convolutional codes with $K = 3$–8. Note that with the $K = 8$

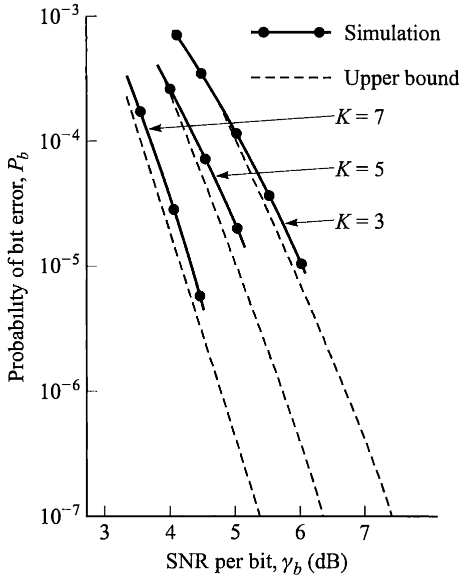


FIGURE 8.6–2
Bit error probability for rate 1/2 Viterbi decoding with eight-level quantized inputs to the decoder and 32-bit path memory. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

code, an error rate of 10^{-5} requires about 6 dB, which represents a coding gain of nearly 4 dB relative to uncoded QPSK.

The effect of input signal quantization is further illustrated in Figure 8.6–4 for a rate 1/2, $K = 5$ code. Note that 3-bit quantization (eight levels) is about 2 dB better than hard-decision decoding, which is the ultimate limit between soft-decision decoding and hard-decision decoding on the AWGN channel. The combined effect of signal quantization and path memory truncation for the rate 1/2, $K = 5$ code with 8-, 16-, and 32-bit path memories and either 1- or 3-bit quantization is shown in Figure 8.6–5. It is apparent from these results that a path memory as short as three constraint lengths does not seriously degrade performance.

When the signal from the demodulator is quantized to more than two levels, another problem that must be considered is the spacing between quantization levels. Figure 8.6–6 illustrates the simulation results for an eight-level uniform quantizer as a function of the quantizer threshold spacing. We observe that there is an optimum

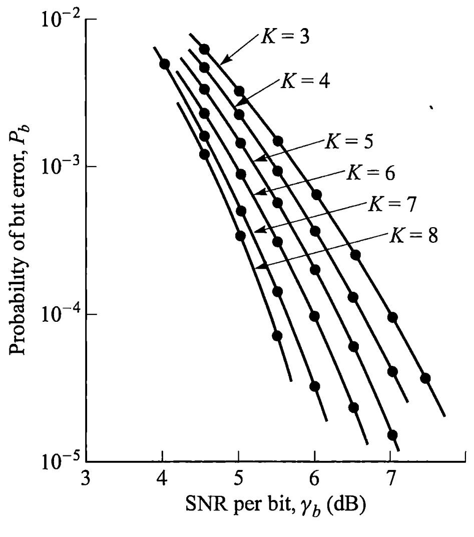


FIGURE 8.6–3
Performance of rate 1/2 codes with hard-decision Viterbi decoding and 32-bit path memory truncation. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

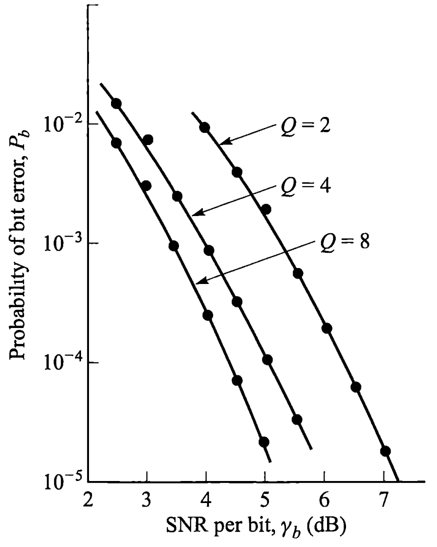


FIGURE 8.6–4
Performance of rate $1/2$, $K = 5$ code with eight-, four-, and two-level quantization at the input to the Viterbi decoder. Path truncation length = 32 bits. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

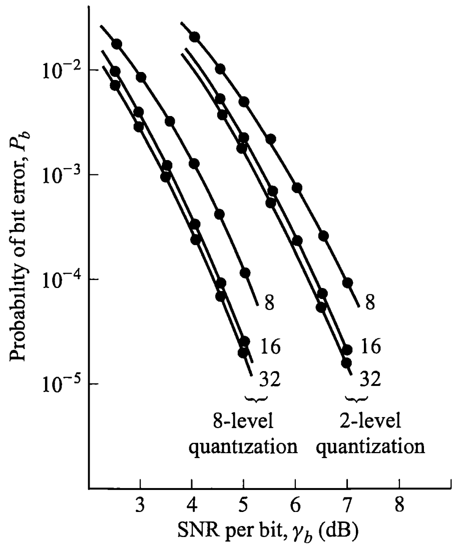


FIGURE 8.6–5
Performance of rate $1/2$, $K = 5$ code with 32-, 16-, and 8-bit path memory truncation and eight- and two-level quantization. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

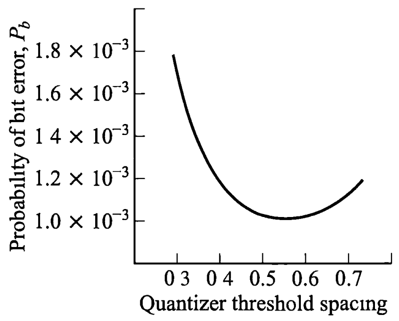


FIGURE 8.6–6
Error rate performance of rate $1/2$, $K = 5$ Viterbi decoder for $\mathcal{E}_b/N_0 = 3.5$ dB and eight-level quantization as a function of quantizer threshold level spacing for equally spaced thresholds. [*From Heller and Jacobs (1971).* © *1971 IEEE.*]

spacing between thresholds (approximately equal to 0.5). However, the optimum is sufficiently broad (0.4–0.7), so that, once it is set, there is little degradation resulting from variations in the AGC level of the order of ±20 percent.

Finally, we should point out some important results in the performance degradation due to carrier phase variations. Figure 8.6–7 illustrates the performance of a rate $1/2$,

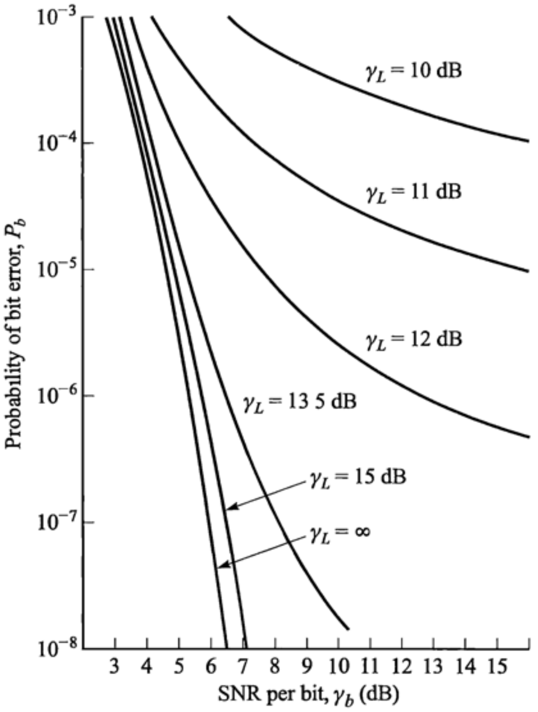


FIGURE 8.6–7
Performance of a rate $1/2$, $K = 7$ code with Viterbi decoding and eight-level quantization as a function of the carrier phase tracking loop SNR γ_L *[From Heller and Jacobs (1971). © 1971 IEEE.]*

$K = 7$ code with eight-level quantization and a carrier phase tracking loop SNR γ_L. Recall that in a PLL, the phase error has a variance that is inversely proportional to γ_L. The results in Figure 8.6–7 indicate that the degradation is large when the loop SNR is small ($\gamma_L < 12$ dB), and causes the error rate performance to bottom out at a relatively high error rate.

8.7
NONBINARY DUAL-*k* CODES AND CONCATENATED CODES

Our treatment of convolutional codes thus far has been focused primarily on binary codes. Binary codes are particularly suitable for channels in which binary or quaternary PSK modulation and coherent demodulation is possible. However, there are many applications in which PSK modulation and coherent demodulation is not suitable or possible. In such cases, other modulation techniques, e.g., M-ary FSK, are employed in conjunction with noncoherent demodulation. Nonbinary codes are particularly matched to M-ary signals that are demodulated noncoherently.

In this subsection, we describe a class of nonbinary convolutional codes, called *dual-k codes*, that are easily decoded by means of the Viterbi algorithm using either soft-decision or hard-decision decoding. They are also suitable either as an outer code or as an inner code in a concatenated code, as will also be described below.

A dual-k rate $1/2$ convolutional encoder may be represented as shown in Figure 8.7–1. It consists of two ($K = 2$) k-bit shift-register stages and $n = 2k$ function generators. Its output is two k-bit symbols. We note that the code considered in Example 8.1–4 is a dual-2 convolutional code.

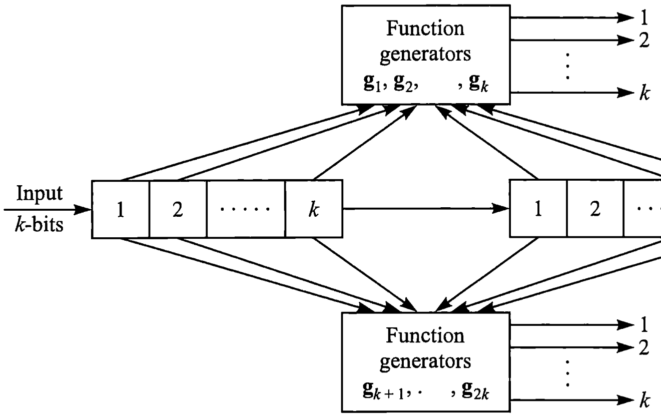

Function generators
$\mathbf{g}_1, \mathbf{g}_2, \quad, \mathbf{g}_k$
1
2
k
Input
k-bits
1
2
·····
k
1
2
·····
k
Function generators
$\mathbf{g}_{k+1}, \cdot \quad, \mathbf{g}_{2k}$
1
2
k

FIGURE 8.7–1
Encoder for rate 1/2 dual-k codes.

The $2k$ function generators for the dual-k codes have been given by Viterbi and Jacobs (1975). These may be expressed in the form

$$\begin{bmatrix} \leftarrow \mathbf{g}_1 \rightarrow \\ \leftarrow \mathbf{g}_2 \rightarrow \\ \vdots \\ \leftarrow \mathbf{g}_k \rightarrow \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 & & \cdots & 0 & 1 \end{bmatrix} = [\mathbf{I}_k \quad \mathbf{I}_k]$$

$$\begin{bmatrix} \leftarrow \mathbf{g}_{k+1} \rightarrow \\ \leftarrow \mathbf{g}_{k+2} \rightarrow \\ \vdots \\ \leftarrow \mathbf{g}_{2k} \rightarrow \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & \cdots & & 0 & 1 & 0 & 0 & & \cdots & & 0 \\ 0 & 0 & 1 & 0 & \cdots & & 0 & 0 & 1 & 0 & & \cdots & & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & & \vdots & \vdots & \vdots & \vdots & & \cdots & & \vdots \\ 0 & 0 & 0 & & \cdots & 0 & 1 & 0 & 0 & & \cdots & & 1 & 0 \\ 1 & 0 & 0 & & \cdots & 0 & 0 & 0 & 0 & & \cdots & & 0 & 1 \end{bmatrix}$$

$$= \left[\begin{matrix} 1 & 1 & 0 & 0 & \cdots & & 0 \\ 0 & 0 & 1 & 0 & \cdots & & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & & \vdots \\ 0 & 0 & 0 & & \cdots & 0 & 1 \\ 1 & 0 & 0 & & \cdots & 0 & 0 \end{matrix} \qquad \mathbf{I}_k \right] \tag{8.7–1}$$

where $\mathbf{I}_k$ denotes the $k \times k$ identity matrix.

The general form for the transfer function of a rate 1/2 dual-k code has been derived by Odenwalder (1976). It is expressed as

$$\begin{aligned} T(Y, Z, J) &= \frac{(2^k - 1)Z^4 J^2 Y}{1 - YJ[2Z + (2^k - 3)Z^2]} \\ &= \sum_{i=4}^{\infty} a_i Z^i Y^{f(i)} J^{h(i)} \end{aligned} \tag{8.7–2}$$

where D represents the Hamming distance for the q-ary ($q = 2^k$) symbols, the $f(i)$ exponent on N represents the number of information symbol errors that are produced in selecting a branch in the tree or trellis other than a corresponding branch on the all-zero path, and the $h(i)$ exponent on J is equal to the number of branches in a given path. Note that the minimum free distance is $d_{\text{free}} = 4$ symbols ($4k$ bits).

Lower-rate dual-k convolutional codes can be generated in a number of ways, the simplest of which is to repeat each symbol generated by the rate 1/2 code r times, where $r = 1, 2, \ldots, m$ ($r = 1$ corresponds to each symbol appearing once). If each symbol in any particular branch of the tree or trellis or state diagram is repeated r times, the effect is to increase the distance parameter from Z to Z^r. Consequently the transfer function for a rate $1/2r$ dual-k code is

$$T(Y, Z, J) = \frac{(2^k - 1)Z^{4r}J^2Y}{1 - YJ[2Z^r + (2^k - 3)Z^{2r}]} \tag{8.7–3}$$

In the transmission of long information sequences, the path length parameter J in the transfer function may be suppressed by setting $J = 1$. The resulting transfer function $T(Y, Z)$ may be differentiated with respect to Y, and Y is set to unity. This yields

$$\begin{aligned} \left.\frac{dT(Y, Z)}{dY}\right|_{N=1} &= \frac{(2^k - 1)Z^{4r}}{[1 - 2Z^r - (2^k - 3)Z^{2r}]^2} \\ &= \sum_{i=4r}^{\infty} \beta_i Z^i \end{aligned} \tag{8.7–4}$$

where β_i represents the number of symbol errors associated with a path having distance Z^i from the all-zero path, as described previously in Section 8.2–2. The expression in Equation 8.7–4 may be used to evaluate the error probability for dual-k codes under various channel conditions.

Performance of dual-k codes with M-ary modulation Suppose that a dual-k code is used in conjunction with M-ary orthogonal signaling at the modulator, where $M = 2^k$. Each symbol from the encoder is mapped into one of the M possible orthogonal waveforms. The channel is assumed to add white Gaussian noise. The demodulator consists of M matched filters.

If the decoder performs hard-decision decoding, the performance of the code is determined by the symbol error probability P_e. This error probability has been computed in Chapter 4 for both coherent and noncoherent detection. From P_e, we can determine $P_2(d)$ according to Equation 8.2–16 or 8.2–17, which is the probability of error in a pairwise comparison of the all-zero path with a path that differs in d symbols. The probability of a bit error is upper-bounded as

$$P_b < \frac{2^{k-1}}{2^k - 1} \sum_{d=4r}^{\infty} \beta_d P_2(d) \tag{8.7–5}$$

The factor $2^{k-1}/(2^k - 1)$ is used to convert the symbol error probability to the bit error probability.

Instead of hard-decision decoding, suppose that the decoder performs soft-decision decoding using the output of a demodulator that employs a square-law detector. The expression for the bit error probability given by Equation 8.7–5 still applies, but now $P_2(d)$ is given by (see Section 11.1–1)

$$P_2(d) = \frac{1}{2^{2d-1}} \exp\left(-\frac{1}{2}\gamma_b R_c d\right) \sum_{i=0}^{d-1} K_i \left(\frac{1}{2}\gamma_b R_c d\right)^i \tag{8.7–6}$$

where

$$K_i = \frac{1}{i!} \sum_{l=0}^{d-1-i} \binom{2d-1}{l} \tag{8.7–7}$$

and $R_c = 1/2r$ is the code rate.

Concatenated codes In Section 7.13–2, we considered the concatenation of two block codes to form a long block code. Now that we have described convolutional codes, we broaden our viewpoint and consider the concatenation of a block code with a convolutional code or the concatenation of two convolutional codes.

In a conventional concatenated code, the outer code is usually chosen to be nonbinary, with each symbol selected from an alphabet of $q = 2^k$ symbols. This code may be a block code, such as a Reed–Solomon code, or a convolutional code, such as a dual-k code. The inner code may be either binary or nonbinary, and either a block or a convolutional code. For example, a Reed–Solomon code may be selected as the outer code and a dual-k code may be selected as the inner code. In such a concatenation scheme, the number of symbols in the outer (Reed–Solomon) code q equals 2^k, so that each symbol of the outer code maps into a k-bit symbol of the inner dual-k code. M-ary orthogonal signals may be used to transmit the symbols.

The decoding of such concatenated codes may also take a variety of different forms. If the inner code is a convolutional code having a short constraint length, the Viterbi algorithm provides an efficient means for decoding, using either soft-decision or hard-decision decoding.

If the inner code is a block code, and the decoder for this code performs soft-decision decoding, the outer decoder may also perform soft-decision decoding using as inputs the metrics corresponding to each word of the inner code. On the other hand, the inner decoder may make a hard decision after receipt of the code word and feed the hard decisions to the outer decoder. Then the outer decoder must perform hard-decision decoding.

The following example describes a concatenated code in which the outer code is a convolutional code and the inner code is a block code.

EXAMPLE 8.7–1. Suppose we construct a concatenated code by selecting a dual-k code as the outer code and a Hadamard code as the inner code. To be specific, we select a rate 1/2 dual-5 code and a Hadamard (16, 5) inner code. The dual-5 rate 1/2 code has a minimum free distance $D_{\text{free}} = 4$ and the Hadamard code has a minimum distance $d_{\min} = 8$. Hence, the concatenated code has an effective minimum distance of 32. Since there are 32 code words in the Hadamard code and 32 possible symbols in the outer code, in effect, each symbol from the outer code is mapped into one of the 32 Hadamard code words.

The probability of a symbol error in decoding the inner code may be determined from the results of the performance of block codes given in Sections 7.4 and 7.5 for soft-decision and hard-decision decoding, respectively. First, suppose that hard-decision decoding is performed in the inner decoder with the probability of a code word (symbol of outer code) error denoted as P_{32}, since $M = 32$. Then the performance of the outer code and, hence, the performance of the concatenated code is obtained by using this error probability in conjunction with the transfer function for the dual-5 code given by Equation 8.7–2.

On the other hand, if soft-decision decoding is used on both the outer and the inner codes, the soft-decision metric from each received Hadamard code word is passed to the Viterbi algorithm, which computes the accumulated metrics for the competing paths through the trellis. We shall give numerical results on the performance of concatenated codes of this type in our discussion of coding for Rayleigh fading channels.

8.8
MAXIMUM A POSTERIORI DECODING OF CONVOLUTIONAL CODES—THE BCJR ALGORITHM

The BCJR algorithm, named after Bahl, Cocke, Jelinek, and Raviv Bahl et al. (1974), is a symbol-by-symbol maximum a posteriori decoding algorithm for convolutional codes. In this algorithm the decoder uses the MAP algorithm to decode each input symbol to the decoder rather than looking for the most likely input sequence.

We know that convolutional codes are finite memory encoders in which the output and the next state depend on the current state and the input. Assuming $k = 1$, we denote an information sequence of length N by $\boldsymbol{u} = (u_1, u_2, \ldots, u_N)$ where $u_i \in \{0, 1\}$, and the corresponding encoded sequence by[†] $\boldsymbol{c} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_N)$ where the length of $\boldsymbol{c}_i$ is n. The encoder state at time i is denoted by σ_i. For $1 \leq i \leq N$ we have

$$\boldsymbol{c}_i = f_c(u_i, \sigma_{i-1}) \tag{8.8–1}$$

$$\sigma_i = f_s(u_i, \sigma_{i-1}) \tag{8.8–2}$$

where functions f_c and f_s define the codeword and the new state as functions of the input $u_i \in \{0, 1\}$ and the previous state $\sigma_{i-1} \in \Sigma$, where Σ denotes the set of all states. It is clear that any pair of states (σ_{i-1}, σ_i) that satisfies Equation 8.8–2 corresponds either to $u_i = 1$ or to $u_i = 0$. Therefore, we can partition the set of all pairs of state (σ_{i-1}, σ_i) which correspond to all possible transitions into two subsets S_0 and S_1, corresponding to $u_i = 0$ and $u_i = 1$, respectively.

The symbol-by-symbol maximum a posteriori decoding receives $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N)$, the demodulator output, and based on this observation decodes u_i using the maximum a posteriori rule

---

[†] We use $\boldsymbol{c}$ to denote both the encoded sequence, which is a binary sequence of length nN with elements from $\{0, 1\}$, and the encoded sequence after BPSK modulation, which is a sequence of length nN with elements from $\pm\sqrt{\mathcal{E}_c}$. It should be clear from the context which notion is used.

$$\begin{aligned}\hat{u}_i &= \arg\max_{u_i\in\{0,1\}} P(u_i|\boldsymbol{y})\\ &= \arg\max_{u_i\in\{0,1\}} \frac{p(u_i,\boldsymbol{y})}{p(\boldsymbol{y})}\\ &= \arg\max_{u_i\in\{0,1\}} p(u_i,\boldsymbol{y})\\ &= \arg\max_{\ell\in\{0,1\}} \sum_{(\sigma_{i-1},\sigma_i)\in S_\ell} p(\sigma_{i-1},\sigma_i,\boldsymbol{y})\end{aligned} \tag{8.8–3}$$

where the last equality follows from the fact that $u_i = l$ corresponds to all pairs of state $(\sigma_{i-1},\sigma_i)\in S_\ell$ for $\ell = 0, 1$.

If we define

$$\begin{aligned}\boldsymbol{y}_1^{(i-1)} &= \left(\boldsymbol{y}_1,\dots,\boldsymbol{y}^{(i-1)}\right)\\ \boldsymbol{y}_{i+1}^{(N)} &= (\boldsymbol{y}_{i+1},\dots,\boldsymbol{y}_N)\end{aligned} \tag{8.8–4}$$

we can write

$$\boldsymbol{y} = \left(\boldsymbol{y}_1^{(i-1)},\boldsymbol{y}_i,\boldsymbol{y}_{i+1}^{(N)}\right) \tag{8.8–5}$$

and we have

$$\begin{aligned}p(\sigma_{i-1},\sigma_i,\boldsymbol{y}) &= p\left(\sigma_{i-1},\sigma_i,\boldsymbol{y}_1^{(i-1)},\boldsymbol{y}_i,\boldsymbol{y}_{i+1}^{(N)}\right)\\ &= p\left(\sigma_{i-1},\sigma_i,\boldsymbol{y}_1^{(i-1)},\boldsymbol{y}_i\right)p\left(\boldsymbol{y}_{i+1}^{(N)}|\sigma_{i-1},\sigma_i,\boldsymbol{y}_1^{(i-1)},\boldsymbol{y}_i\right)\\ &= p\left(\sigma_{i-1},\boldsymbol{y}_1^{(i-1)}\right)p\left(\sigma_i,\boldsymbol{y}_i|\sigma_{i-1},\boldsymbol{y}_1^{(i-1)}\right)p\left(\boldsymbol{y}_{i+1}^{(N)}|\sigma_{i-1},\sigma_i,\boldsymbol{y}_1^{(i-1)},\boldsymbol{y}_i\right)\\ &= p\left(\sigma_{i-1},\boldsymbol{y}_1^{(i-1)}\right)p(\sigma_i,\boldsymbol{y}_i|\sigma_{i-1})\,p\left(\boldsymbol{y}_{i+1}^{(N)}|\sigma_i\right)\end{aligned} \tag{8.8–6}$$

where the first three steps follow from the chain rule and the last step follows from Markov properties of the state in a trellis.

At this point we define $\alpha_{i-1}(\sigma_{i-1})$, $\beta_i(\sigma_i)$, and $\gamma_i(\sigma_{i-1},\sigma_i)$ as

$$\begin{aligned}\alpha_{i-1}(\sigma_{i-1}) &= p\left(\sigma_{i-1},\boldsymbol{y}_1^{(i-1)}\right)\\ \beta_i(\sigma_i) &= p\left(\boldsymbol{y}_{i+1}^{(N)}|\sigma_i\right)\\ \gamma_i(\sigma_{i-1},\sigma_i) &= p(\sigma_i,\boldsymbol{y}_i|\sigma_{i-1})\end{aligned} \tag{8.8–7}$$

Using these definitions in Equation 8.8–6, we have

$$p(\sigma_{i-1},\sigma_i,\boldsymbol{y}) = \alpha_{i-1}(\sigma_{i-1})\,\gamma_i(\sigma_{i-1},\sigma_i)\,\beta_i(\sigma_i) \tag{8.8–8}$$

and hence from Equation 8.8–3 we obtain

$$\hat{u}_i = \arg\max_{\ell\in\{0,1\}} \sum_{(\sigma_{i-1},\sigma_i)\in S_\ell} \alpha_{i-1}(\sigma_{i-1})\,\gamma_i(\sigma_{i-1},\sigma_i)\,\beta_i(\sigma_i) \tag{8.8–9}$$

Equation 8.8–9 indicates that for maximum a posteriori decoding we need the values of $\alpha_{i-1}(\sigma_{i-1})$, $\beta_i(\sigma_i)$, and $\gamma_i(\sigma_{i-1}, \sigma_i)$. It should also be clear that although our development of these equations was based on the assumption of $k = 1$ and $u_i \in \{0, 1\}$, the extension of these results to general k is straightforward.

Now we derive recursion relations for $\alpha_{i-1}(\sigma_{i-1})$ and $\beta_i(\sigma_i)$ which facilitate their computation.

The Forward Recursion for $\alpha_i(\sigma_i)$ We show that $\alpha_{i-1}(\sigma_{i-1})$ can be obtained by using a *forward recursion* of the form

$$\alpha_i(\sigma_i) = \sum_{\sigma_{i-1}\in\Sigma} \gamma_i(\sigma_{i-1}, \sigma_i)\,\alpha_{i-1}(\sigma_{i-1}), \qquad 1 \le i \le N \tag{8.8–10}$$

To prove Equation 8.8–10, we use the following set of relations

$$\begin{aligned}
\alpha_i(\sigma_i) &= p\left(\sigma_i, \boldsymbol{y}_1^{(i)}\right)\\
&= \sum_{\sigma_{i-1}\in\Sigma} p\left(\sigma_{i-1}, \sigma_i, \boldsymbol{y}_1^{(i-1)}, \boldsymbol{y}_i\right)\\
&= \sum_{\sigma_{i-1}\in\Sigma} p\left(\sigma_{i-1}, \boldsymbol{y}_1^{(i-1)}\right) p\left(\sigma_i, \boldsymbol{y}_i|\sigma_{i-1}, \boldsymbol{y}_1^{(i-1)}\right)\\
&= \sum_{\sigma_{i-1}\in\Sigma} p\left(\sigma_{i-1}, \boldsymbol{y}_1^{(i-1)}\right) p\left(\sigma_i, \boldsymbol{y}_i|\sigma_{i-1}\right)\\
&= \sum_{\sigma_{i-1}\in\Sigma} \alpha_{i-1}(\sigma_{i-1})\,\gamma_i(\sigma_{i-1}, \sigma_i)
\end{aligned} \tag{8.8–11}$$

which completes the proof of the forward recursion relation for $\alpha_i(\sigma_i)$. This relation means that given the values of $\gamma_i(\sigma_{i-1}, \sigma_i)$, it is possible to obtain $\alpha_i(\sigma_i)$ from $\alpha_{i-1}(\sigma_{i-1})$. If we assume that the trellis starts in the all-zero state, the initial condition for the forward recursion becomes

$$\alpha_0(\sigma_0) = P(\sigma_0) = \begin{cases} 1 & \sigma_0 = 0 \\ 0 & \sigma_0 \neq 0 \end{cases} \tag{8.8–12}$$

Equations 8.8–10 and 8.8–12 provide a complete set of recursions for computing the values of α.

The Backward Recursion for $\beta_i(\sigma_i)$ The *backward recursion* for computing the values of β is given by

$$\beta_{i-1}(\sigma_{i-1}) = \sum_{\sigma_i\in\Sigma} \beta_i(\sigma_i)\,\gamma_i(\sigma_{i-1}, \sigma_i), \qquad 1 \le i \le N \tag{8.8–13}$$

To prove this recursion, we note that

$$\begin{aligned}\beta_{i-1}(\sigma_{i-1}) &= p\left(\boldsymbol{y}_i^{(N)}\,|\sigma_{i-1}\right)\\ &= \sum_{\sigma_i\in\Sigma} p\left(\boldsymbol{y}_i, \boldsymbol{y}_{i+1}^{(N)}, \sigma_i\,|\sigma_{i-1}\right)\\ &= \sum_{\sigma_i\in\Sigma} p(\sigma_i, \boldsymbol{y}_i|\sigma_{i-1})\, p\left(\boldsymbol{y}_{i+1}^{(N)}\,|\sigma_i, \boldsymbol{y}_i, \sigma_{i-1}\right)\\ &= \sum_{\sigma_i\in\Sigma} p(\sigma_i, \boldsymbol{y}_i\,|\sigma_{i-1})\, p\left(\boldsymbol{y}_{i+1}^{(N)}\,|\sigma_i\right)\\ &= \sum_{\sigma_i\in\Sigma} \gamma_i(\sigma_{i-1}, \sigma_i)\, \beta_i(\sigma_i)\end{aligned} \tag{8.8–14}$$

The boundary condition for the backward recursion, assuming that the trellis is terminated in the all-zero state, is

$$\beta_N(\sigma_N) = \begin{cases} 1 & \sigma_N = 0 \\ 0 & \sigma_N \neq 0 \end{cases} \tag{8.8–15}$$

The recursive relations 8.8–10 and 8.8–13 together with initial conditions 8.8–12 and 8.8–15 provide the necessary equations to determine α's and β's when γ's are known. We now focus on computation of γ's.

Computing $\boldsymbol{\gamma_i(\sigma_{i-1}, \sigma_i)}$ We can write $\gamma_i(\sigma_{i-1}, \sigma_i)$, $1 \leq i \leq N$, as

$$\begin{aligned}\gamma_i(\sigma_{i-1}, \sigma_i) &= p(\sigma_i, \boldsymbol{y}_i|\sigma_{i-1})\\ &= p(\sigma_i|\sigma_{i-1})\, p(\boldsymbol{y}_i|\sigma_i, \sigma_{i-1})\\ &= P(u_i)p(\boldsymbol{y}_i|u_i)\\ &= P(u_i)p(\boldsymbol{y}_i|\boldsymbol{c}_i)\end{aligned} \tag{8.8–16}$$

where we have used the fact that there exists a one-to-one correspondence between a pair of states (σ_{i-1}, σ_i) and the input u_i through Equation 8.8–2. The above expression clearly shows the dependence of $\gamma_i(\sigma_{i-1}, \sigma_i)$ on $P(u_i)$, the prior probability of the information sequence at time i, as well as $p(\boldsymbol{y}_i|\boldsymbol{c}_i)$ which depends on the channel characteristics. If the information sequence is equiprobable, an assumption that is usually made when no information is available, then $P(u_i = 0) = P(u_i = 1) = \frac{1}{2}$. Obviously, the above derivation is based on the assumption that the state pair (σ_{i-1}, σ_i) is a valid pair; i.e., a transition from σ_{i-1} to σ_i is possible.

Equation 8.8–9 together with the forward and backward relations for α and β given in Equations 8.8–10 and 8.8–13 and Equation 8.8–16 for γ are known as the BCJR algorithm for symbol-by-symbol MAP decoding of a convolutional code.

Note that unlike the Viterbi algorithm that looks for the most likely information sequence, the BCJR finds the most likely individual bits, or symbols. The BCJR algorithm also provides the values of $P(u_i|\boldsymbol{y})$. These values provide a level of certainty of the decoder about the value of u_i and are called *soft outputs* or soft values. Having

$P\,(u_i|\boldsymbol{y})$, we can find the a posteriori L values as

$$\begin{aligned} L(u_i) &= \ln \frac{P\,(u_i = 1|\boldsymbol{y})}{P\,(u_i = 0|\boldsymbol{y})} \\ &= \ln \frac{P\,(u_i = 1, \boldsymbol{y})}{P\,(u_i = 0, \boldsymbol{y})} \\ &= \ln \frac{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_1} \alpha_{i-1}\,(\sigma_{i-1})\,\gamma_i\,(\sigma_{i-1},\sigma_i)\,\beta_i\,(\sigma_i)}{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_0} \alpha_{i-1}\,(\sigma_{i-1})\,\gamma_i\,(\sigma_{i-1},\sigma_i)\,\beta_i\,(\sigma_i)} \end{aligned} \tag{8.8–17}$$

which are also referred to as soft outputs. Knowledge of soft outputs is crucial in decoding of turbo codes discussed later in this chapter. A decoder such as the BCJR decoder that accepts soft inputs (the vector $\boldsymbol{y}$) and generates soft outputs is called a *soft-input soft-output* (SISO) decoder. Note that the decoding rule based on $L(u_i)$ soft values is given by

$$\hat{u}_i = \begin{cases} 1 & L(u_i) \geq 0 \\ 0 & L(u_i) < 0 \end{cases} \tag{8.8–18}$$

For an AWGN channel, $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{n}$, where $\boldsymbol{c}$ represents the modulated signal corresponding to the encoded sequence, we have

$$\gamma_i\,(\sigma_{i-1},\sigma_i) = \frac{P(u_i)}{(\pi N_0)^{n/2}} \exp\left(-\frac{\|\boldsymbol{y}_i - \boldsymbol{c}_i\|^2}{N_0}\right) \tag{8.8–19}$$

EXAMPLE 8.8–1. Let us consider the special case when $n = 2$, the convolutional code is systematic, and the modulation is BPSK. In this case we have $\boldsymbol{c}_i = \left(c_i^s, c_i^p\right)$ and $\boldsymbol{y}_i = \left(y_i^s, y_i^p\right)$, where the superscripts s and p represent the terms corresponding to the systematic (information) bit and parity check bit, respectively. Here $c_i^s = \pm\sqrt{\mathcal{E}_c}$ depending on whether $u_i = 1$ or $u_i = 0$. The value of c_i^p can also be one of the two possible values of $\pm\sqrt{\mathcal{E}_c}$. Using these values, Equation 8.8–19 becomes

$$\begin{aligned} \gamma_i\,(\sigma_{i-1},\sigma_i) &= \frac{P(u_i)}{\pi N_0} \exp\left(-\frac{\left(y_i^s - c_i^s\right)^2 + \left(y_i^p - c_i^p\right)^2}{N_0}\right) \\ &= \frac{1}{\pi N_0} \exp\left\{-\frac{\left(y_i^s\right)^2 + \left(y_i^p\right)^2 + 2\mathcal{E}_c}{N_0}\right\} P(u_i) \exp\left(\frac{2y_i^s c_i^s + 2y_i^p c_i^p}{N_0}\right) \end{aligned} \tag{8.8–20}$$

Note that the term $\frac{1}{\pi N_0} \exp\left\{-\frac{(y_i^s)^2+(y_i^p)^2+2\mathcal{E}_c}{N_0}\right\}$ in Equation 8.8–20 is independent of u_i and hence is canceled from the numerator and the denominator of the a posteriori L values in Equation 8.8–17. It is also clear that in the numerator of Equation 8.8–17, which corresponds to $u_i = 1$, we have $c_i^s = \sqrt{\mathcal{E}_c}$ and in the denominator $c_i^s = -\sqrt{\mathcal{E}_c}$.

In this case the a posteriori L values simplify as

$$L(u_i) = \ln \frac{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_1} \alpha_{i-1}(\sigma_{i-1})\, P(u_i) \exp\left(\frac{2y_i^s c_i^s + 2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)}{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_0} \alpha_{i-1}(\sigma_{i-1})\, P(u_i) \exp\left(\frac{2y_i^s c_i^s + 2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)}$$

$$= \frac{4\sqrt{\mathcal{E}_c}\, y_i^s}{N_0} + \ln \frac{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_1} \alpha_{i-1}(\sigma_{i-1})\, P(u_i) \exp\left(\frac{2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)}{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_0} \alpha_{i-1}(\sigma_{i-1})\, P(u_i) \exp\left(\frac{2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)}$$

$$= \frac{4\sqrt{\mathcal{E}_c}\, y_i^s}{N_0} + \ln \frac{P(u_i = 1)}{P(u_i = 0)} + \ln \frac{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_1} \alpha_{i-1}(\sigma_{i-1}) \exp\left(\frac{2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)}{\displaystyle\sum_{(\sigma_{i-1},\sigma_i)\in S_0} \alpha_{i-1}(\sigma_{i-1}) \exp\left(\frac{2y_i^p c_i^p}{N_0}\right) \beta_i(\sigma_i)} \tag{8.8–21}$$

One problem with the version of the BCJR algorithm described above is that it is not a numerically stable algorithm, particularly if the trellis length is long. An alternative to this algorithm is the log-domain version of it known as the Log-APP (log a posteriori probability) algorithm.†

In the Log-APP algorithm, instead of α, β, and γ, we define their logarithms as

$$\begin{aligned} \widetilde{\alpha}_i(\sigma_i) &= \ln(\alpha_i(\sigma_i)) \\ \widetilde{\beta}_i(\sigma_i) &= \ln(\beta_i(\sigma_i)) \\ \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i) &= \ln(\gamma_i(\sigma_{i-1},\sigma_i)) \end{aligned} \tag{8.8–22}$$

Straightforward calculation shows the following forward and backward recursions hold for $\widetilde{\alpha}_i(\sigma_i)$ and $\widetilde{\beta}_i(\sigma_{i-1})$.

$$\begin{aligned} \widetilde{\alpha}_i(\sigma_i) &= \ln\left(\sum_{\sigma_{i-1}\in\Sigma} \exp\left(\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i)\right)\right) \\ \widetilde{\beta}_{i-1}(\sigma_{i-1}) &= \ln\left(\sum_{\sigma_i\in\Sigma} \exp\left(\widetilde{\beta}_i(\sigma_i) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i)\right)\right) \end{aligned} \tag{8.8–23}$$

with initial conditions

$$\widetilde{\alpha}_0(\sigma_0) = \begin{cases} 0 & \sigma_0 = 0 \\ -\infty & \sigma_0 \neq 0 \end{cases} \qquad \widetilde{\beta}_N(\sigma_N) = \begin{cases} 0 & \sigma_N = 0 \\ -\infty & \sigma_N \neq 0 \end{cases} \tag{8.8–24}$$

---

†Also called Log-MAP algorithm.

and the a posteriori L values are computed as

$$\begin{aligned} L(u_i) &= \ln \left[\sum_{(\sigma_{i-1},\sigma_i)\in S_1} \exp(\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i) + \widetilde{\beta}_i(\sigma_i)) \right] \\ &\quad - \ln \left[\sum_{(\sigma_{i-1},\sigma_i)\in S_0} \exp(\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i) + \widetilde{\beta}_i(\sigma_i)) \right] \end{aligned} \tag{8.8–25}$$

These relations are numerically more stable but are not computationally efficient. To improve the computational efficiency, we can introduce the following notation:

$$\begin{aligned} \max{}^*\{x, y\} &\triangleq \ln(e^x + e^y) \\ \max{}^*\{x, y, z\} &\triangleq \ln(e^x + e^y + e^z) \end{aligned} \tag{8.8–26}$$

Using these definitions, we have the recursions

$$\begin{aligned} \widetilde{\alpha}_i(\sigma_i) &= \max_{\sigma_{i-1}\in\Sigma}{}^* \{\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i)\} \\ \widetilde{\beta}_{i-1}(\sigma_{i-1}) &= \max_{\sigma_i\in\Sigma}{}^* \{\widetilde{\beta}_i(\sigma_i) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i)\} \end{aligned} \tag{8.8–27}$$

where the initial conditions for these recursions are given by Equation 8.8–24. The a posteriori L values are given by

$$\begin{aligned} L(u_i) = &\max_{(\sigma_{i-1},\sigma_i)\in S_1}{}^* \{\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i) + \widetilde{\beta}_i(\sigma_i)\} \\ - &\max_{(\sigma_{i-1},\sigma_i)\in S_0}{}^* \{\widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \widetilde{\gamma}_i(\sigma_{i-1},\sigma_i) + \widetilde{\beta}_i(\sigma_i)\} \end{aligned} \tag{8.8–28}$$

The initial conditions for these recursions are given by Equation 8.8–24.

EXAMPLE 8.8–2. For the special case studied in Example 8.8–1, the expression for the a posteriori L values can be obtained using the log-domain quantities in Equation 8.8–21. The result is

$$\begin{aligned} L(u_i) = \frac{4\sqrt{\mathcal{E}_c}\, y_i^s}{N_0} + L^a(u_i) + &\max_{(\sigma_{i-1},\sigma_i)\in S_1}{}^* \left\{ \widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \frac{2y_i^p c_i^p}{N_0} + \widetilde{\beta}_i(\sigma_i) \right\} \\ - &\max_{(\sigma_{i-1},\sigma_i)\in S_0}{}^* \left\{ \widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \frac{2y_i^p c_i^p}{N_0} + \widetilde{\beta}_i(\sigma_i) \right\} \end{aligned} \tag{8.8–29}$$

where we have defined $L^a(u_i)$ as

$$L^a(u_i) = \ln \frac{P(u_i = 1)}{P(u_i = 0)} \tag{8.8–30}$$

It is seen that in this case the a posteriori L values can be written as the sum of three terms. The first term, $\frac{4\sqrt{\mathcal{E}_c}\, y_i^s}{N_0}$, depends on the channel output corresponding to the systematic bits received by the decoder. The second term, $L^a(u_i)$, depends on the a priori probabilities of the information bits. The remaining term is the contribution of the channel outputs corresponding to the parity bits.

It can be easily shown that (Problem 8.22)

$$\begin{aligned} \max{}^*\{x, y\} &= \max\{x, y\} + \ln\left(1 + e^{-|x-y|}\right) \\ \max{}^*\{x, y, z\} &= \max{}^*\left\{\max{}^*\{x, y\}, z\right\} \end{aligned} \tag{8.8–31}$$

The term $\ln\left(1 + e^{-|x-y|}\right)$ is small when x and y are not close. Its maximum occurs when $x = y$ for which this term is $\ln 2$. It is clear that for large x and y or when x and y are not close, we can use the approximation

$$\max{}^*\{x, y\} \approx \max\{x, y\} \tag{8.8–32}$$

Under similar conditions we can use the approximation

$$\max{}^*\{x, y, z\} \approx \max\{x, y, z\} \tag{8.8–33}$$

The approximate relations in Equations 8.8–32 and 8.8–33 are valid when the values of x and y (or x, y, and z) are not close. In general, approximating $\max^*$ by max in Equation 8.8–27 would result in a small performance degradation. The resulting algorithm, which is a suboptimal implementation of the MAP algorithm, is called that Max-Log-APP algorithm.[†]

Instead of using the approximations given in Equations 8.8–32 and 8.8–33, one can use a lookup table for values of the correction term $\ln\left(1 + e^{-|x-y|}\right)$ to improve the performance. The interested reader is referred to Robertson and Hoeher (1997), Ryan (2003), Robertson et al. (1995), and Lin and Costello (2004) for details.

8.9

TURBO CODES AND ITERATIVE DECODING

In Section 7.13–2 we introduced serial and parallel concatenated block codes in which an interleaver is used to construct extremely long codes. In this section we consider the construction and decoding of concatenated codes with interleaving, using convolutional codes.

Parallel concatenated convolutional codes (PCCCs) with interleaving, also called *turbo codes*, were introduced by Berrou et al. (1993) and Berrou and Glavieux (1996). A basic turbo encoder, shown in Figure 8.9–1, is a recursive systematic encoder that employs two recursive systematic convolutional encoders in parallel, where the second encoder is preceded by an interleaver. The two recursive systematic convolutional encoders may be either identical or different. We observe that the nominal rate at the output of the turbo encoder is $R_c = 1/3$. However, by puncturing the parity check bits at the output of the binary convolutional encoders, we may achieve higher rates, such as rate 1/2 or 2/3. As in the case of concatenated block codes, the interleaver is usually selected to be a block pseudorandom interleaver that reorders the bits in the information sequence before feeding them to the second encoder. In effect, as will be shown later,

---

[†] Also called Max-Log-MAP algorithm.

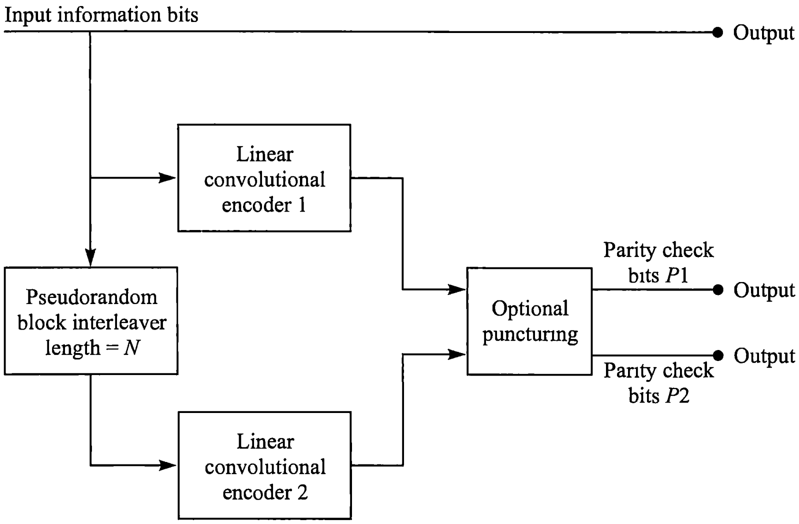

Input information bits
Output
Linear convolutional encoder 1
Pseudorandom block interleaver length = N
Optional puncturing
Parity check bits $P1$
Output
Output
Parity check bits $P2$
Linear convolutional encoder 2

FIGURE 8.9–1
Encoder for parallel concatenated code (turbo code).

the use of two recursive convolutional encoders in conjunction with the interleaver produces a code that contains very few codewords of low weight. This characteristic does not necessarily imply that the free distance of the concatenated code is especially large. However, the use of the interleaver in conjunction with the two encoders results in codewords that have relatively few nearest neighbors. That is, the codewords are relatively sparse. Hence, the coding gain achieved by a turbo code is due in part to this feature, i.e., the reduction in the number of nearest-neighboring codewords, called the *multiplicity*, that result from interleaving.

A standard turbo code shown in Figure 8.9–1 is completely described by the constituent codes, which are usually similar, and the interleaving pattern, usually denoted by Π. The constituent codes, being recursive and systematic, are given by their generator matrix of the form

$$G(D) = \begin{bmatrix} 1 & \frac{g_2(D)}{g_1(D)} \end{bmatrix} \tag{8.9–1}$$

where $g_1(D)$ and $g_2(D)$ specify the feedback and the feedforward connections, respectively. Usually the constituent codes are specified by the octal representation of g_1 and g_2.

EXAMPLE 8.9–1. A (31, 27) RSC encoder is represented by $\boldsymbol{g}_1 = (11001)$ and $\boldsymbol{g}_2 = (10111)$ corresponding to $g_1(D) = 1 + D + D^4$ and $g_2(D) = 1 + D^2 + D^3 + D^4$. The encoder is given by the block diagram shown in Figure 8.9–2.

8.9–1 Performance Bounds for Turbo Codes

Turbo codes are two recursive systematic convolutional codes concatenated by an interleaver. Although the codes are linear and time-invariant, the operation of the interleaver, although linear, is not time-invariant. The trellis of the resulting linear but time-varying

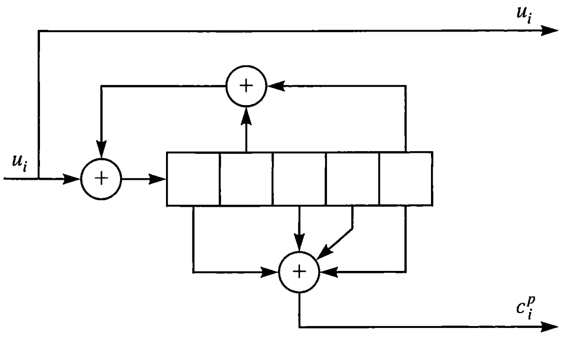


FIGURE 8.9–2
A (31, 27) RSC encoder.

finite-state machine has a huge number of states that makes maximum-likelihood decoding hopeless. In Benedetto and Montorsi (1996) it is stated that a certain turbo code that has been implemented in VLSI when viewed as a time-varying finite-state machine has 2^{1030} states, making maximum-likelihood decoding impractical.

Although maximum-likelihood decoding of turbo codes is impractical, it can serve to find an upper bound on the performance of these codes. By linearity of turbo codes, we can assume that the all-zero information sequence is transmitted. Assuming an interleaver of length N, there exist a total of 2^N possible information sequences with weights between 0 (for the all-zero sequence) and N. Let $m \in \{1, 2, \ldots, 2^N - 1\}$ denote the erroneous information sequence that is detected when the all-zero sequence is transmitted, and let us denote the weight of this sequence by j_m, where $1 \leq j_m \leq N$. Note that since the code is systematic, the weight of the codeword corresponding to the information sequence m, denoted by w_m, is the sum of the weight of the information sequence j_m and the weight of the corresponding parity sequence. The probability of decoding m when the all-zero sequence is transmitted, assuming BPSK modulation, is given by

$$P_{\mathbf{0}\to m} = Q\left(\sqrt{2R_c w_m \gamma_b}\right) \tag{8.9–2}$$

and the corresponding bit error probability when m is detected is given by

$$P_b(\mathbf{0} \to m) = \frac{j_m}{N} Q\left(\sqrt{2R_c w_m \gamma_b}\right) \tag{8.9–3}$$

Using the union bound, the average bit error probability is bounded by

$$P_b \leq \frac{1}{N} \sum_{m=1}^{2^N-1} j_m Q\left(\sqrt{2R_c w_m \gamma_b}\right) \tag{8.9–4}$$

Reordering and grouping the terms corresponding to information sequences of the same weight, we can write

$$P_b \leq \frac{1}{N} \sum_{j=1}^{N} \sum_{l=1}^{\binom{N}{j}} jQ\left(\sqrt{2R_c d_{jl} \gamma_b}\right) \tag{8.9–5}$$

where $\binom{N}{j}$ is the number of information sequences of weight j and d_{jl} is the weight of the codeword generated by the lth information sequence of weight j. Now let us consider the following cases as applied to the PCCC shown in Figure 8.9–1.

Information Sequences of Weight $j = 1$ An information sequence with weight 1 ($j = 1$) when applied to a recursive convolutional code generates the impulse response of the convolutional code. Since recursive convolutional codes have infinite impulse response, or very large weight impulse response even when they are terminated, the case of $j = 1$ results in large values for d_{jl} and thus very low bit error probability. The only case that can cause a problem occurs when the single 1 in the input sequence occurs at the end of a block of length N, in which case the output weight is low. The existence of the pseudorandom interleaver, however, makes it highly unlikely that after interleaving the single 1 will not appear at the end of the block and thus would generate a high-weight codeword when applied to the second encoder. The probability of having a single 1 at the end of the block both before and after interleaving is very small.

Information Sequences of Weight $j = 2$ There exist $\binom{N}{2}$ information sequences of weight 2 corresponding to polynomials of the form $D^{i_1} + D^{i_2} = D^{i_1}(D^{i_2-i_1} + 1)$, where $0 \leq i_1 < i_2 \leq N - 1$, and i_1 and i_2 determine the location of the 1s in the information sequence. In general, a polynomial of this form when applied to $g_2(D)/g_1(D)$ generates parity symbols of large weight, unless $g_1(D)$ divides $D^\ell + 1$, where $\ell = i_2 - i_1$. If this is the case, then $D^\ell + 1 = g_1(D)h(D)$, where $h(D)$ is a polynomial. The parity sequences generated by $D^{i_1} + D^{i_2}$ in this case will be $D^{i_1}h(D)g_2(D)$ which can correspond to a low-weight parity sequence. For instance, if $g_1(D) = 1 + D + D^2$, then $g_1(D)$ divides any weight 2 sequence of the form $D^{i_1}(D^3 + 1)$, resulting in a parity polynomial of the form $D^{i_1}(1 + D)g_2(D)$ which can correspond to a parity sequence of low weight. In this example any information sequence of weight 2 in which there are two zeros between the two 1s will result in a low-weight parity sequence.[†] The existence of the interleaver, however, makes it highly unlikely that an information sequence of weight 2 would generate low-weight parity sequences both before and after interleaving. In fact, the number of weight 2 information sequences that generate low-weight parity polynomials before and after interleaving is much smaller than N, where N is the interleaver length. In contrast, for a single RSCC this number is of the order of N.

A similar argument can be applied to weight 3 and weight 4 information sequences. In both cases it can be argued that due to the effect of the interleaver, the number of weight 3 and weight 4 information sequences that generate low-weight parities is much lower than N. This means that low-weight codewords are possible in turbo codes, but their occurence is very low. In other words, the main factor contributing to the excellent performance of turbo codes particularly at low signal-to-noise ratios is not their good distance structure, but the relatively low multiplicity of codewords with low weight. Note that the effect of low multiplicity of turbo codes is particularly noticeable at low signal-to-noise ratios. At higher signal-to-noise ratios, the low minimum distance of these codes results in an *error floor*.

If we consider information sequences of weight 2 and 3 as the main contributors to the error probability bound for turbo codes, we can approximate the bit error bound of Equation 8.9–5 as

---

[†] Obviously, this also applies to the case where there are five zeros between two 1s, etc.

$$P_b \lesssim \frac{1}{N} \sum_{j=2}^{3} j n_j Q\left(\sqrt{2R_c d_{j,\min} \gamma_b}\right) \tag{8.9–6}$$

where $d_{j,\min}$ denotes the minimum codeword weight among all codewords generated by information sequences of weight j and $n_j \ll N$ denotes the number of information sequences of weight j that generate codewords of weight $d_{j,\min}$. Since $n_j \ll N$, the coefficient of $Q\left(\sqrt{2R_c d_{j,\min} \gamma_b}\right)$ is much smaller than 1. The effect of the factor $1/N$ that drastically reduces the error bound on turbo codes is called the *interleaver gain.*

The bounds discussed above are based on the union bounding technique that is loose particularly at low signal-to-noise ratios. More advanced bounding techniques have been studied and applied to turbo codes that provide tighter bounds at low signal-to-noise ratios. The interested reader is referred to Duman and Salehi (1997), Sason and Shamai (2000), and Sason and Shamai (2001b).

8.9–2 Iterative Decoding for Turbo Codes

We have seen that optimal decoding of turbo codes is impossible due to the large number of states in the code trellis. A suboptimal iterative decoding algorithm, known as the *turbo decoding algorithm,* was proposed by Berrou et al. (1993) which achieves excellent performance very close to the theoretical bound predicted by Shannon.

The turbo decoding algorithm is based on iterative usage of the Log-APP or the Max-Log-APP algorithm. As it was shown in Example 8.8–2, the a posteriori L values can be written as the sum of three terms as

$$L(u_i) = L_c y_i^s + L^{(a)}(u_i) + L^{(e)}(u_i) \tag{8.9–7}$$

where

$$\begin{aligned}
L_c y_i^s &= \frac{4\sqrt{\mathcal{E}_c}\, y_i^s}{N_0} \\
L^a(u_i) &= \ln \frac{P(u_i = 1)}{P(u_i = 0)} \\
L^{(e)}(u_i) &= \max_{(\sigma_{i-1},\sigma_i)\in S_1}{}^{*} \left\{ \widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \frac{2y_i^p c_i^p}{N_0} + \widetilde{\beta}_i(\sigma_i) \right\} \\
&\quad - \max_{(\sigma_{i-1},\sigma_i)\in S_0}{}^{*} \left\{ \widetilde{\alpha}_{i-1}(\sigma_{i-1}) + \frac{2y_i^p c_i^p}{N_0} + \widetilde{\beta}_i(\sigma_i) \right\}
\end{aligned} \tag{8.9–8}$$

and we have defined $L_c = \frac{4}{N_0}\sqrt{\mathcal{E}_c}$.

The term $L_c y_i^s$ is called the *channel L value* and denotes the effect of channel outputs corresponding to the systematic bits. The second term $L^a(u_i)$ is the *a priori L value* and is a function of the a priori probabilities of the information sequence. The final term, $L^{(e)}(u_i)$, represents the *extrinsic L value* or *extrinsic information* which is the part of the a posteriori L value that does not depend on the a priori probabilities and the systematic information at the channel output.

Let us assume that the binary information sequence $\boldsymbol{u} = (u_1, u_2, \ldots, u_N)$ is applied to the first rate 1/2 RSCC, and let us denote the parity bits at the output by $\boldsymbol{c}^p = (c_1^p, c_2^p, \ldots, c_N^p)$. The information sequence is passed through the interleaver to obtain $\boldsymbol{u}' = (u_1', u_2', \ldots, u_N')$, and this sequence is applied to the second encoder to generate the parity sequence $\boldsymbol{c}'^p = (c_1'^p, c_2'^p, \ldots, c_N'^p)$. Sequences $\boldsymbol{u}$, $\boldsymbol{c}^p$, and $\boldsymbol{c}'^p$ are BPSK modulated and transmitted over a Gaussian channel. The corresponding output sequences are denoted by $\boldsymbol{y}^s$, $\boldsymbol{y}^e$, and $\boldsymbol{y}'^p$. The MAP decoder for the first constituent code receives the pair $(\boldsymbol{y}^s, \boldsymbol{y}^p)$. In the first iteration the decoder assumes all bits are equiprobable, and therefore the a priori L values are set to zero. Having access to $(\boldsymbol{y}^s, \boldsymbol{y}^p)$, the first decoder uses Equation 8.8–29 to compute the a posteriori L values. At the output of the first constituent decoder, the decoder subtracts the channel L values from the a posteriori L values to compute the extrinsic L values. These values are denoted by $L_{12}^{(e)}(u_i)$ and are permuted by the interleaver Π and then used by the second constituent decoder as its a priori L values. In addition to this information, the second decoder is supplied with $\boldsymbol{y}'^p$ and a permuted version of $\boldsymbol{y}^s$ after passing it through the interleaver Π. The second decoder computes the extrinsic L values denoted by $L_{21}^{(e)}(u_i)$ and after permuting them through Π^{-1} supplies them to the first encoder, which in the next iteration uses these values as its a priori L values. This process is continued either for a fixed number of iterations or until a certain criterion is met. After the last iteration the a posteriori L values $L(u_i)$ are used to make the final decision.

The building block of the turbo decoder is an SISO decoder with inputs $\boldsymbol{y}^s$, $\boldsymbol{y}^p$, and $L^{(a)}(u_i)$ and outputs $L^{(e)}(u_i)$ and $L(u_i)$. In iterative decoding $L^{(a)}(u_i)$ is substituted by the extrinsic L values provided by the other decoder. The block diagram of a turbo decoder is shown in Figure 8.9–3.

A typical plot of the performance of the iterative decoding algorithm for turbo codes is given in Figure 8.9–4. It is clearly seen that the first few iterations noticeably improve the performance. It is seen from these plots that three regions are distinguishable. For the low-SNR region where the error probability changes very slowly as a function of $\mathcal{E}_b/N_0$ and the number of iterations, for moderate SNRs the error probability drops rapidly with increasing $\mathcal{E}_b/N_0$ and over many iterations P_b decreases consistently. This region is called the *waterfall region* or the *turbo cliff region*. Finally, for moderately large $\mathcal{E}_b/N_0$ values, the code exhibits an error floor which is typically achieved with a

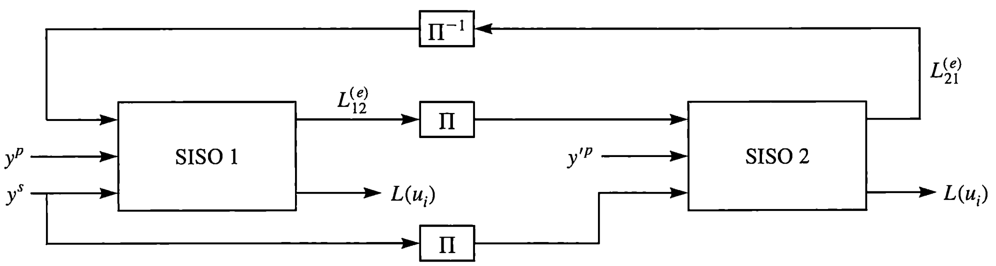


FIGURE 8.9–3
Block diagram of a turbo decoder.

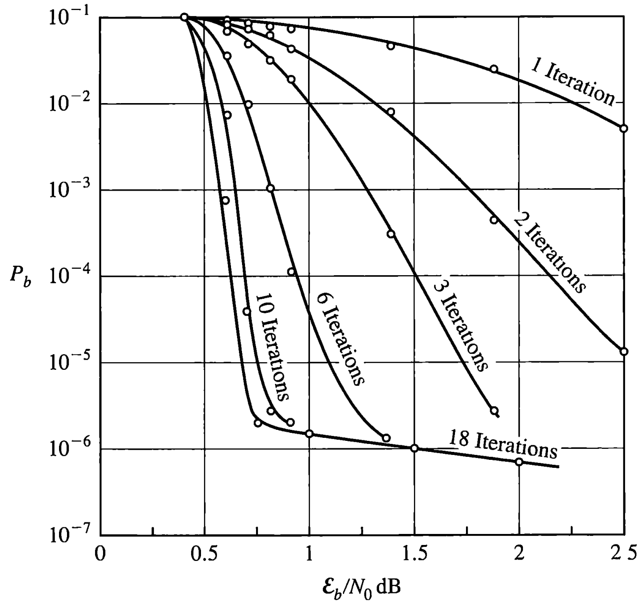


FIGURE 8.9–4
Performance of iterative decoding for turbo codes.

few iterations. As discussed before, the error floor effect in turbo codes is due to their low minimum distance.

Typically, four iterations are adequate if the decoders are operating at a high enough SNR to achieve an error rate in the range 10^{-5} to 10^{-6}, whereas about eight to ten iterations may be needed when the error rate is in the range of 10^{-5}, where the SNR is lower.

An important factor in the performance of the turbo code is the length of the interleaver, which is sometimes referred to as the *interleaver gain.* With a sufficiently large interleaver and iterative MAP decoding, the performance of a turbo code is very close to the Shannon limit. For example, a rate $1/2$ turbo code of block length $N = 2^{16}$ with 18 iterations of decoding per bit achieves an error probability of 10^{-5} at an SNR of 0.7 dB. From Figure 6.5–6 we see that the Shannon limit for a binary input rate $1/2$ code is roughly 0.19 dB. This means that this code operates 0.5 dB from the Shannon limit.

The major drawback with decoding turbo codes with large interleavers is the decoding delay and the computational complexity inherent in the iterative decoding algorithm. In most data communication systems, however, the decoding delay is tolerable, and the additional computational complexity is usually justified by the significant coding gain that is achieved by the turbo code. A second method for constructing concatenated convolutional codes with interleaving is serial concatenation. Benedetto et al. (1998) have investigated the construction and the performance of serial concatenated convolutional codes (SCCCs) with interleaving and have developed an iterative decoding algorithm for such codes. In comparing the error rate performance of SCCC with PCCC (turbo codes), Benedetto et al. (1998) found that SCCC generally exhibit better performance than PCCC for error rates below 10^{-2}. For more details on the properties of turbo codes, the reader is referred to Lin and Costello (2004), Benedetto and Montorsi (1996), Heegard and Wicker (1999), and Hagenauer et al. (1996).

8.9–3 EXIT Chart Study of Iterative Decoding

Due to complexity of the iterative decoding algorithm, study of its convergence properties is difficult. A useful tool in studying the performance of iterative decoding of turbo codes, particularly in the turbo cliff region, is the *Extrinsic Information Transfer* (EXIT) chart. These charts were introduced by ten Brink (2001) and have served as a useful tool in performance study and design of different iterative algorithms.

In Section 8.9–2 we have seen that an iterative decoder for a standard turbo code consists of two similar SISO decoders which accept the a priori and channel information at their input and generate the extrinsic information and the log-likelihood values at the output. The two SISO decoders are connected in such a way that the extrinsic information $L^{(e)}$ of each serves as the a priori information $L^{(a)}$ for the other one. The development of the EXIT chart is based on the empirical observation (ten Brink (2001)) that the a priori L value and the transmitted systematic bits are related through

$$L^{(a)} = \frac{\sigma^2}{2} C^{(s)} + n_a \qquad (8.9\text{–}9)$$

where n_a is a zero-mean Gaussian random variable with variance σ^2, and $C^{(s)}$ denotes the normalized systematic transmitted symbol that can take values ± 1. From this we conclude that

$$p_{L^{(a)}|C^{(s)}}(\ell|c) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\left(\ell - c\sigma^2/2\right)^2}{2\sigma^2}} \qquad (8.9\text{–}10)$$

where $c = \pm 1$ with equal probability. The mutual information between $L^{(a)}$ and $C^{(s)}$ is denoted by I_a and is given by

$$I_a = \frac{1}{2} \sum_{c=-1,1} \int_{-\infty}^{\infty} p(\ell|c) \log_2 \frac{2p(\ell|c)}{p(\ell|C=-1) + p(\ell|C=1)} \, d\ell \qquad (8.9\text{–}11)$$

Using Equation 8.9–10 in 8.9–11 and using an approach similar to the approach taken in the derivation of Equations 6.5–31 and 6.5–32, we obtain

$$I_a = 1 - \mathrm{E}\left[\log_2\left(1 + e^{-C^{(s)} \cdot L^{(a)}}\right)\right] \qquad (8.9\text{–}12)$$

where the expectation is with respect to the joint distribution of $C^{(s)}$ and $L^{(a)}$.

It is clear that $0 \le I_a \le 1$, and it can be shown to be a monotonically increasing function of σ; thus given the value of I_a, σ can be uniquely determined.

A similar argument can be applied to the extrinsic information $L^{(e)}$ to derive I_e, the mutual information between $L^{(e)}$ and $C^{(s)}$. The extrinsic information transfer (EXIT) characteristic is defined as I_e when expressed as a function of I_a and $\mathcal{E}_b/N_0$, i.e.,

$$I_e = T(I_e, \mathcal{E}_b/N_0) \qquad (8.9\text{–}13)$$

or simply as

$$I_e = T(I_a) \qquad (8.9\text{–}14)$$

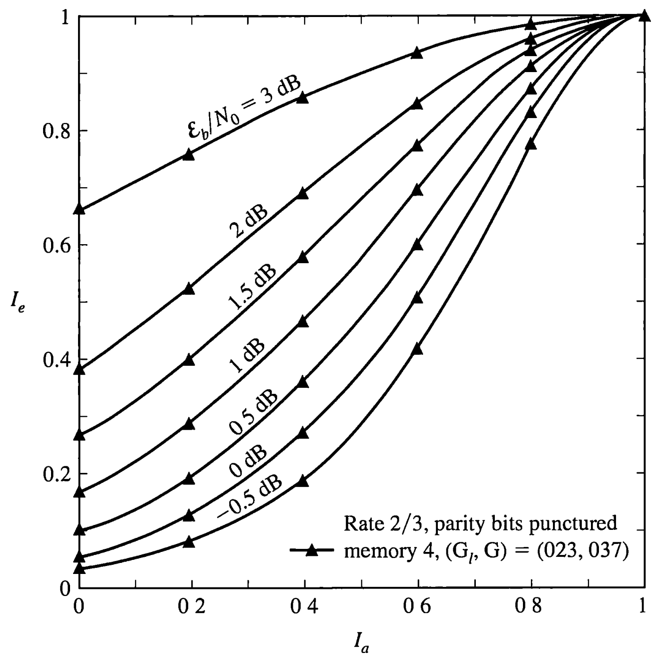


FIGURE 8.9–5
EXIT chart for a rate 2/3 convolutional code for different values of $\mathcal{E}_b/N_0$. [*From ten Brink (2001) © IEEE.*]

where this characteristic is plotted for different values of $\mathcal{E}_b/N_0$. Since the values of I_a and I_e are not given explicitly, Monte Carlo simulation is usually used to find the expected value in Equation 8.9–12. This is done over a large number of samples N, and I_a is computed as

$$I_a \approx 1 - \frac{1}{N}\sum_{n=1}^{N}\log_2\left(1 + e^{-c_n \ell_n}\right) \tag{8.9–15}$$

The EXIT chart for a (23, 37) RSCC after puncturing to increase the rate from 1/2 to 2/3 is shown in Figure 8.9–5. The plots are shown for values of $\mathcal{E}_b/N_0$ in the range of −0.5 dB to 3 dB.

For turbo codes, the extrinsic information generated by a decoder acts as the a priori information for the next stage. To study the operation of an iterative decoder for a turbo code, we plot the two EXIT functions of the constituent codes and move between the two plots along the horizontal and vertical directions corresponding to equating the extrinsic information of one encoder to the a priori information of the other, as shown in Figure 8.9–6.

As seen in Figure 8.9–6, the iterative decoding begins with the assumption of equal probabilities for the information bits. This corresponds to $I_{a1} = 0$ and moves horizontally and vertically between the two EXIT graphs. It is seen that when $\mathcal{E}_b/N_0 = 0.1$ dB, the two EXIT graphs intersect at low values of I_a and I_e, as noted in the lower left corner of Figure 8.9–6. In this case after a couple of iterations no more improvement is achieved, and low values of mutual information indicate a high error probability. This behavior corresponds to the low signal-to-noise ratio region in Figure 8.9–4 and sometimes is referred to as the *pinch-off region*. For higher values of $\mathcal{E}_b/N_0$, the two EXIT graphs become separated and there exists a *bottleneck region* through which the iterative decoding trajectory climbs to high I_a and I_e values corresponding to low error

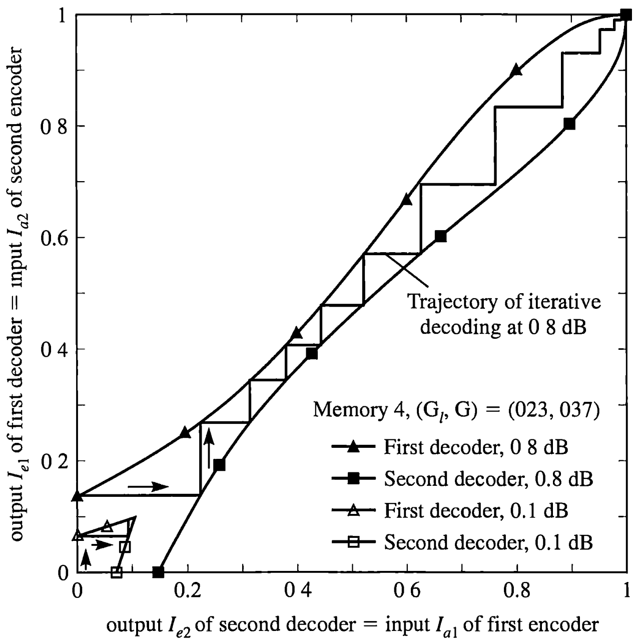


FIGURE 8.9–6
Simulated trajectories of iterative decoding for $\mathcal{E}_b/N_0 = 0.1$ and $\mathcal{E}_b/N_0 = 0.8$ dB. [*From ten Brink (2001) © IEEE.*]

probabilities. This region corresponds to the *waterfall region* in Figure 8.9–4. Finally, for large $\mathcal{E}_b/N_0$ values the graphs in the EXIT charts become wide open with fast convergence to the error floor. Figure 8.9–7 depicts another example of EXIT charts for various values of $\mathcal{E}_b/N_0$. The trajectories for $\mathcal{E}_b/N_0 = 0.7$ dB corresponding to the waterfall region and $\mathcal{E}_b/N_0 = 1.5$ dB are shown for comparison.

In addition to providing insight to the performance of iterative decoding schemes, EXIT charts have been used in the design of highly efficient codes as well as other iterative methods such as iterative equalization.

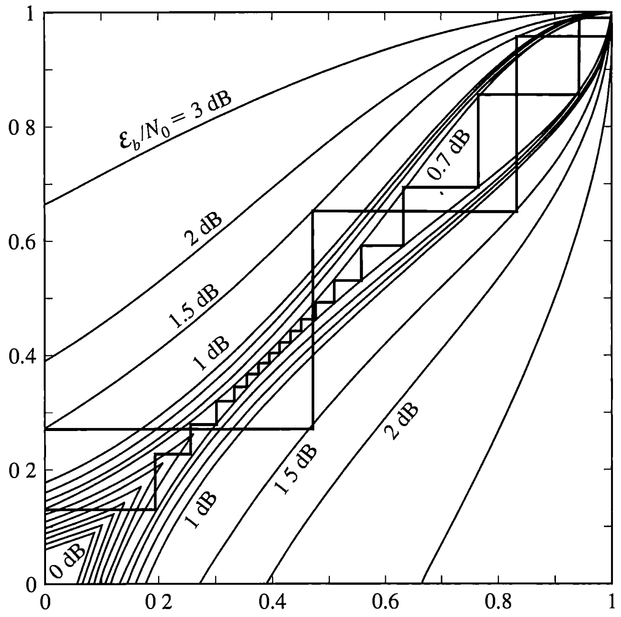


FIGURE 8.9–7
EXIT chart trajectories for $\mathcal{E}_b/N_0 = 0.7$ dB and $\mathcal{E}_b/N_0 = 1.5$ dB. Simulation is done for an interleaver size of 10^6 bits. [*From ten Brink (2001) © IEEE.*]

8.10
FACTOR GRAPHS AND THE SUM-PRODUCT ALGORITHM

We have observed that the trellis representation of convolutional codes is a convenient graphical representation that is very useful in the implementation and understanding of the maximum-likelihood decoding of these codes using the Viterbi algorithm or the symbol-by-symbol maximum a posteriori decoding using the BCJR algorithm. Representation of codes by more general graphical models is a convenient method in studying the performance of some decoding algorithms. Graph representation is not limited to decoding algorithms but has many applications to signal processing, circuit theory, control theory, networking, and probability theory. In this section we provide an introductory treatment of some of the basic graphical models used in the design of a general algorithm called the *sum-product algorithm*.

The sum-product algorithm was first introduced by Gallager (1963) as a decoding method for *low-density parity check* (LDPC) codes. Later, Tanner (1981) introduced graphical models to describe this class of codes. These graphical models are known as *Tanner graphs*. Wiberg et al. (1995) and Wiberg (1996) showed that the Viterbi and BCJR algorithms as well as decoding algorithms for turbo and LDPC codes can be unified in a single algorithm on certain graphs. The idea of graph representation of codes was further developed and generalized by Forney (2001).

8.10–1 Tanner Graphs

Recall that an (n, k) linear block code $\mathcal{C}$ is described by a $k \times n$ generator matrix $\boldsymbol{G}$ through

$$\boldsymbol{c} = \boldsymbol{u}\boldsymbol{G} \tag{8.10–1}$$

where $\boldsymbol{u}$ is an information sequence of length k and $\boldsymbol{c}$ is the corresponding codeword. A binary sequence of length n is a codeword of $\mathcal{C}$ if and only if Equation 8.10–1 is satisfied for some binary sequence $\boldsymbol{u}$. The parity check matrix of this code $\boldsymbol{H}$ is an $(n-k) \times n$ binary matrix defined as the generator matrix of the dual code $\mathcal{C}^{\perp}$. A necessary and sufficient condition for $\boldsymbol{c}$ to be a codeword is that

$$\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0} \tag{8.10–2}$$

This equation can be written in terms of $n-k$ relations

$$\begin{aligned} \boldsymbol{c}\boldsymbol{h}_1^t &= 0 \\ \boldsymbol{c}\boldsymbol{h}_2^t &= 0 \\ \vdots &= \vdots \\ \boldsymbol{c}\boldsymbol{h}_{n-k}^t &= 0 \end{aligned} \tag{8.10–3}$$

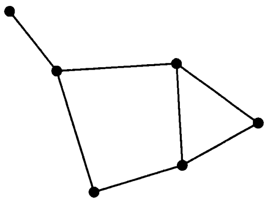


FIGURE 8.10–1
An example of a graph.

where $\boldsymbol{h}_i$ denotes the ith row of $\boldsymbol{H}$. These equations introduce a set of $n - k$ linear constraints on a codeword $\boldsymbol{c}$. For instance in a (7, 4) Hamming code with

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{8.10–4}$$

these equations become

$$\begin{aligned} c_1 + c_2 + c_3 + c_5 &= 0 \\ c_2 + c_3 + c_4 + c_6 &= 0 \\ c_1 + c_2 + c_4 + c_7 &= 0 \end{aligned} \tag{8.10–5}$$

where addition is modulo-2. For a (3, 1) repetition code we have

$$\boldsymbol{H} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} \tag{8.10–6}$$

and the parity check equations become

$$\begin{aligned} c_1 + c_2 &= 0 \\ c_1 + c_3 &= 0 \end{aligned} \tag{8.10–7}$$

A Tanner graph is a graphical representation of Equations 8.10–3 as a *bipartite graph*. In general, a graph is a collection of nodes (or vertices) and edges (or links) such that each edge connects two nodes; i.e., each edge of the graph is uniquely determined by the two nodes it connects. An example of a graph is shown in Figure 8.10–1. The *degree* of a node is the number of edges that are incident on that node.

A graph is called a *bipartite graph* if the nodes of the graph can be partitioned into two subsets N_1 and N_2 such that each edge has one node in N_1 and one node in N_2. In other words, there exists no edge that connects two nodes both in N_1 or both in N_2. An example of a bipartite graph is shown in Figure 8.10–2.

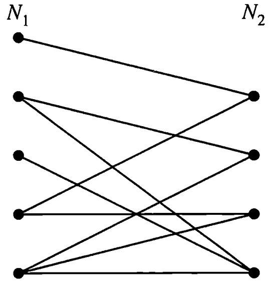


FIGURE 8.10–2
A bipartite graph.

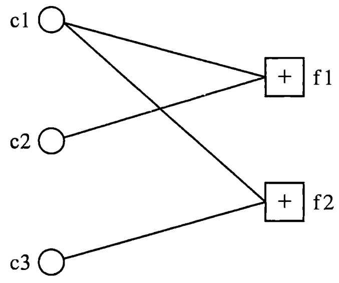


FIGURE 8.10–3
The Tanner graph for the (3, 1) repetition code.

A Tanner graph representation of Equations 8.10–3 can be obtained by representing the each codeword component c_i, $1 \leq i \leq n$, of a codeword $\boldsymbol{c}$ as a node i in N_1 and each of the $n - k$ constraints given by Equation 8.10–3 as a node j, $1 \leq j \leq n - k$, in N_2. There exists an edge connecting node i in N_1 to node j in N_2 if and only if c_i appears in the jth parity check equation. Figures 8.10–3 and 8.10–4 depict the Tanner graphs for the (3, 1) repetition code and the (7, 4) Hamming code, respectively. Note that since $\boldsymbol{H}$ for a code is not unique, its Tanner graph is not unique either.

One major difference between the two graphs shown in Figures 8.10–3 and 8.10–4 is that the first graph does not include *cycles*; that is, a path on the edges does not exist that starts from a node and ends in the same node. However, the second graph includes cycles, as clearly seen on the graph. A *cycle-free graph* is a graph in which removing any edge divides the graph into two disconnected graphs. The length of the shortest cycle included in a graph is called the *girth* of the graph. The girth of the graph shown in Figure 8.10–4 is 4.

In the Tanner graph of Figure 8.10–4 two types of nodes are distinguishable: the *variable nodes*, which correspond to the variables supplied to the Tanner graph (these are

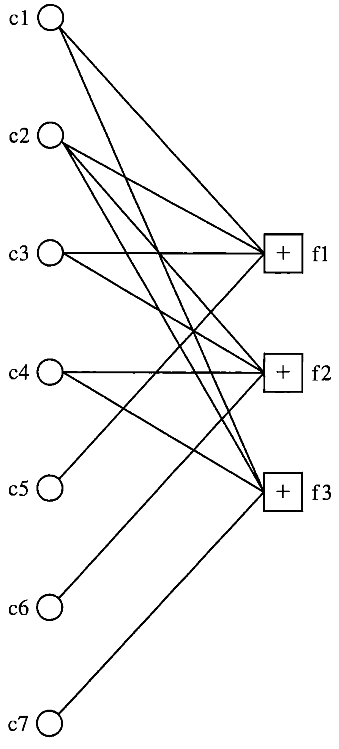


FIGURE 8.10–4
The Tanner graph for the (7, 4) Hamming code.

the nodes denoted by circles on the left), and the *constraint nodes* that force a relation between the variables. These nodes are denoted by squares on the right. A binary sequence $\boldsymbol{c}$ is a codeword if it satisfies the three constraints given by Equations 8.10–5. Let us define the indicator function of a proposition P as

$$\delta[P] = \begin{cases} 1 & \text{if } P \text{ is true} \\ 0 & \text{if } P \text{ is false} \end{cases} \tag{8.10–8}$$

Then, for instance,

$$\delta[c_1 + c_2 + c_3 + c_5 = 0] = \begin{cases} 1 & \text{if } c_1 + c_2 + c_3 + c_5 = 0 \\ 0 & \text{if } c_1 + c_2 + c_3 + c_5 = 1 \end{cases} \tag{8.10–9}$$

and $\boldsymbol{c}$ is a codeword if

$$\delta[c_1+c_2+c_3+c_5 = 0]\delta[c_2+c_3+c_4+c_6 = 0]\delta[c_1+c_2+c_4+c_7 = 0] = 1 \tag{8.10–10}$$

The graph shown in Figure 8.10–4 is a graphical representation of the relation given by Equation 8.10–10. We note that the product function of Equation 8.10–10 which represents a global constraint for $\boldsymbol{c}$ to be a codeword can be factored into three local constraints. Any input to this graph is a valid input if it results in a nonzero global value for the global equation of the graph; and this can occur only if the input is a codeword. Tanner graphs are special cases of factor graphs to be studied in the next section.

8.10–2 Factor Graphs

Let us assume that $f(x_1, x_2, \ldots, x_n)$ is a real-valued function of n variables $x_1, \ldots, x_n$ where x_i takes values in a discrete set $\mathcal{X}$. Assume we are interested in computing a *marginal* function of one variable $f_i(x_i)$ as

$$f_i(x_i) = \sum_{x_1}\sum_{x_2}\cdots\sum_{x_{i-1}}\sum_{x_{i+1}}\cdots\sum_{x_n} f(x_1, x_2, \ldots, x_n) \tag{8.10–11}$$

This, for instance, can be the case if we have the joint PDF of n random variables and want to compute the marginal PDF of x_i. If the size of the set $\mathcal{X}$ is $|\mathcal{X}|$, then computing this sum requires $|\mathcal{X}|^{n-1}$ operations. If we use the the shorthand notation $\sim x_i$ to indicate summation over all variables except x_i, then Equation 8.10–11 can be written in the more compact form

$$f_i(x_i) = \sum_{\sim x_i} f(x_1, \ldots, x_n) \tag{8.10–12}$$

Computation of $f_i(x_i)$ can be made considerably simpler if the *global function* $f(x_1, x_2, \ldots, x_n)$ is a factor of some *local functions* depending on a subset of variables, i.e., if for $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ we can write

$$f(\boldsymbol{x}) = \prod_{m=1}^{M} g_m(\boldsymbol{x}_m) \tag{8.10–13}$$

where $\boldsymbol{x}_m$, $1 \leq m \leq M$, is a subset of components of $\boldsymbol{x}$. For instance, in the case where

$$f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8) = g_1(x_1)g_2(x_2)g_3(x_1, x_2, x_3, x_4)g_4(x_4, x_5, x_6) \\ \times g_5(x_5)g_6(x_6, x_7, x_8)g_7(x_7) \tag{8.10–14}$$

we have

$$f_4(x_4) = \left(\sum_{x_1,x_2,x_3} g_1(x_1)g_2(x_2)g_3(x_1, x_2, x_3, x_4)\right) \\ \times \left(\sum_{x_5,x_6} g_4(x_4, x_5, x_6)g_5(x_5)\left(\sum_{x_7,x_8} g_6(x_6, x_7, x_8)g_7(x_7)\right)\right) \tag{8.10–15}$$

which requires less computation than the general case.

Let us assume that $f(\boldsymbol{x})$ is given by Equation 8.10–13. Then a factor graph representing this global function is a graph consisting of a M nodes and n edge or half-edges. An edge connects two nodes, and a half-edge just represents a value entering a node. Therefore a half-edge on one side is connected to a node and on the other side is free. Each edge or half-edge of the factor graph uniquely represents a variable, and each node uniquely represents a local function. Since we are assuming that each edge or half-edge uniquely represents a variable, this representation is possible only if each variable appears in at most two local functions. We will see shortly how this limitation can be removed.

EXAMPLE 8.10–1. The factor graph representing

$$p(w, u, v, x_1, x_2, y) = p(u, v, w)p(x_1|u)p(x_2|v)p(y|x_1, x_2) \tag{8.10–16}$$

is shown in Figure 8.10–5. Note that two half edges corresponding to variables w and y appear just in one local function.

If a variable appears in more than two local functions, we introduce a cloning node that makes copies of this variable. Then we can supply these copies to local functions (nodes on the graph) that need them. A cloning node is given by equality constraints.

EXAMPLE 8.10–2. Let us consider the function

$$f(x_1, x_2, x_3, x_4, x_5) = g_1(x_1, x_2)g_2(x_1, x_3)g_3(x_1, x_4)g_4(x_3, x_4, x_5) \tag{8.10–17}$$

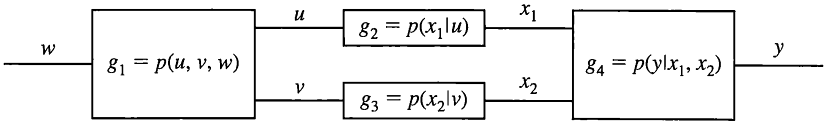


FIGURE 8.10–5
Factor graph representing Equation 8.10–16.

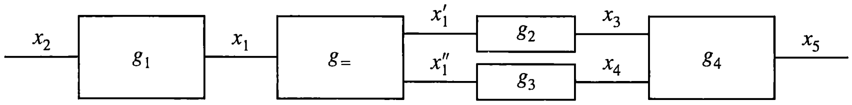


FIGURE 8.10–6
The factor graph representing Equation 8.10–17.

In this function the variable x_1 appears in three local functions and hence has to be cloned. The factor graph in Figure 8.10–6 shows how the equality constraint is introduced to carry out this cloning. The equality constraint is a local function of the form

$$g_=(x_1, x_1', x_1'') = \delta(x_1 = x_1')\delta(x_1 = x_1'') \tag{8.10–18}$$

This means that the value of this local function is 1 if and only if $x_1 = x_1' = x_1''$. If this constraint is not satisfied, the value of the function is zero, making the value of the global function zero. This means that for such values of (x_1, x_1', x_1'') the value of the global function is not positive, and hence such a combination is not a valid input. Introducing $g_=$ as in Equation 8.10–18 makes it possible to have a variable in more than two local functions.

EXAMPLE 8.10–3. The factor graph representation of the Tanner graph for the Hamming code shown in Figure 8.10–4 is shown in Figure 8.10–7.

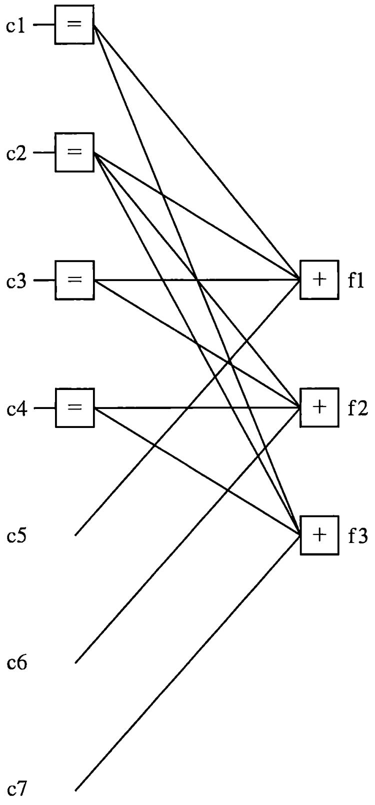


FIGURE 8.10–7
The factor graph representation for a (7, 4) Hamming code.

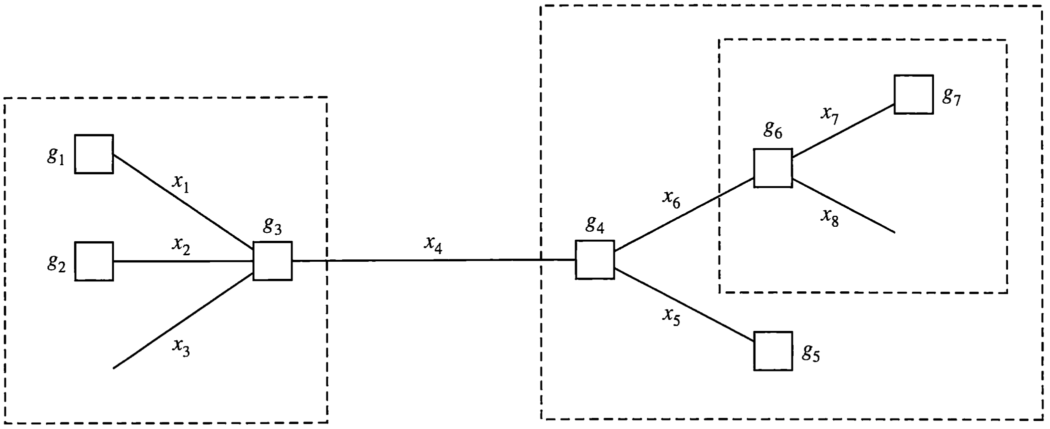


FIGURE 8.10–8
The factor graph representation of the function in Equation 8.10–14.

8.10–3 The Sum-Product Algorithm

The sum-product algorithm is an efficient algorithm for computing marginals of the form

$$f(x_i) = \sum_{\sim x_i} f(x_1, x_2, \ldots, x_n) \tag{8.10–19}$$

using the factor graph for $f(x_1, \ldots, x_n)$. The basic idea is to sum over some of the variables and then transmit two different messages in opposite directions across each edge of the factor graph. The messages transmitted across each edge are functions of the variable corresponding to that edge. These functions are usually expressed as vectors whose components represent different values that these functions can take for different values of the edge variable. This means that the dimensionality of the vector for each edge is equal to the cardinality of the variable represented by that edge. In applications of this algorithm to coding problems, since variables are usually binary, the vectors representing the messages are two-dimensional vectors. A more convenient way in this case, where the messages usually represent the probabilities of the variable being equal to 0 or 1, is to use the ratio of the probabilities (likelihood ratio) or its logarithm (the log-likelihood ratio LLR).

Let us consider the marginal represented by Equation 8.10–15 as[†]

$$\begin{aligned} f_4(x_4) &= \left(\sum_{x_1,x_2,x_3} g_1(x_1)g_2(x_2)g_3(x_1,x_2,x_3,x_4) \right) \\ &\quad \times \left(\sum_{x_5,x_6} g_4(x_4,x_5,x_6)g_5(x_5) \left(\sum_{x_7,x_8} g_6(x_6,x_7,x_8)g_7(x_7) \right) \right) \end{aligned} \tag{8.10–20}$$

The factor graph for $f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8)$ is represented by Figure 8.10–8, where elements in the boxes correspond to the partial sums in Equation 8.10–20.

[†] This example is taken from Loeliger (2004).

We define

$$\begin{aligned}
\mu_{g_3 \to x_4}(x_4) &= \sum_{x_1, x_2, x_3} g_1(x_1) g_2(x_2) g_3(x_1, x_2, x_3, x_4) \\
\mu_{g_6 \to x_6}(x_6) &= \sum_{x_7, x_8} g_6(x_6, x_7, x_8) g_7(x_7) \\
\mu_{g_4 \to x_4}(x_4) &= \sum_{x_5, x_6} g_4(x_4, x_5, x_6) g_5(x_5) \mu_{g_6 \to x_6}(x_6)
\end{aligned} \tag{8.10–21}$$

as the messages passed at g_3, g_6, and g_4, respectively. Referring to Figure 8.10–8, we note that $\mu_{g_6 \to x_6}(x_6)$ is the message passed out of the inner box summarizing its content and $\mu_{g_3 \to x_4}$ and $\mu_{g_4 \to x_4}$ are the two messages sent in opposite directions on the edge corresponding to variable x_4. Equation 8.10–20 states that the marginal $f_4(x_4)$ is the product of the two messages passed along the edge corresponding to x_4. What we have done here is that we have successively summarized each subsystem and used the result to summarize the next system. The resulting algorithm, known as the sum-product algorithm, can be summarized as follows. Each node corresponding to local function $g(x_1, x_2, \ldots, x_n)$ receives messages corresponding to local variables x_i on the branches corresponding to these variables. The received messages are denoted by $\mu_{x_i \to g}(x_i)$. Based on these messages the node computes the outgoing message $\mu_{g \to x_i}(x_i)$ and sends it over the branch corresponding to x_i. A diagram representing this process is shown in Figure 8.10–9.

The outgoing messages are computed using the relation

$$\mu_{g \to x_i}(x_i) = \sum_{\sim x_i} g(x_1, \ldots, x_n) \prod_{j \neq i} \mu_{x_j \to g}(x_j) \tag{8.10–22}$$

where $\mu_{x_j \to g}(x_j)$ is the incoming message on edge j corresponding to variable x_j. Note that in computing the outgoing message on the edge corresponding to x_i, we have used all incoming messages except the message corresponding to edge x_i. This is equivalent to saying that the extrinsic information is passed over node x_i. For some special nodes the following rules are followed:

1. The message sent over a half-edge to the (single) node connecting to it is a message with value 1.

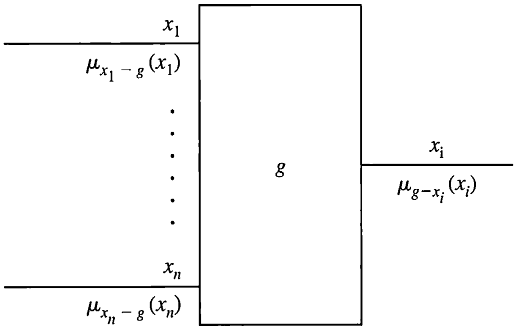


FIGURE 8.10–9
The local computation in sum-product algorithm.

2. If g is a function of a single variable x_i, then the product term in Equation 8.10–22 becomes empty and the equation reduces to

$$\mu_{g \to x_i}(x_i) = g(x_i) \tag{8.10–23}$$

3. For a cloning node $g_=$ with equality constraint, simple substitution in Equation 8.10–22 yields

$$\mu_{g_= \to x_i}(x_i) = \prod_{j \neq i} \mu_{x_j \to g_=}(x_i) \tag{8.10–24}$$

There exists a sharp contrast in applying the sum-product algorithm to cycle-free graphs and graphs with cycles. In a cycle-free graph, the sum-product algorithm can start from all leaves of the graph and proceed along the nodes as their incoming messages become available. Since the graph is cycle-free, each message is computed only once. After this step is done, the marginals corresponding to each variable can be found as the product of the two messages sent in opposite directions on the edge corresponding to that variable. For cycle-free graphs the sum-product algorithm converges to the correct marginals in a finite number of steps. If the graph has cycles, then the convergence of the algorithm is not guaranteed. However, in many practical cases of interest the algorithm converges even for graphs that include cycles.

Factor Graph of a Code

For a code $\mathcal{C}$ with codewords $\boldsymbol{c}_i$, $1 \leq i \leq M$, the global function can be written as $\delta[\boldsymbol{c} \in \mathcal{C}]$. If $\boldsymbol{c}$ is a codeword, then this function is equal to 1, indicating that $\boldsymbol{c}$ is a valid input. For a noncodeword sequence, the value of the global variable is zero, indicating that the input is not valid.

Depending on the code characteristics this global function can be factorized differently. For instance, for convolutional codes this function can be written as the product of the conditions that each component of $\boldsymbol{c}$ must be part of a path through the code trellis and, therefore, must correspond to a transition between states σ_{i-1} and σ_i. For the (7, 4) Hamming code the global function can be written as the product of three parity check (local) functions as

$$\delta[\boldsymbol{c} \in \mathcal{C}] = \delta[c_1 + c_2 + c_3 + c_5 = 0]\delta[c_2 + c_3 + c_4 + c_6 = 0]\delta[c_1 + c_2 + c_4 + c_7 = 0] \tag{8.10–25}$$

In binary block codes two types of nodes are present in the factor graph of the code: the $n - k$ constraint nodes that represent the $n - k$ parity check equations of the form $\boldsymbol{c}\boldsymbol{h}_s^t = 0$ for $1 \leq s \leq n - k$ and the equality constraint nodes (cloning nodes) corresponding to codeword components that appear in more than two parity check equations. We have already seen that for the equality constraint nodes

$$\mu_{g_= \to c_i}(c_i) = \prod_{j \neq i} \mu_{c_j \to g_=}(c_i) \tag{8.10–26}$$

For the parity check nodes, if the messages are two-dimensional vectors representing the probability of the edge variable being[†] 0 or 1, we can show that (see Problem 8.25)

$$\begin{aligned}\mu_{g\to x_i}(c_i = 0) &= \frac{1}{2} + \frac{1}{2}\prod_{j\neq i}(1 - 2p_j(1)) \\ \mu_{g\to x_i}(c_i = 1) &= \frac{1}{2} - \frac{1}{2}\prod_{j\neq i}(1 - 2p_j(1))\end{aligned} \tag{8.10–27}$$

where $p_j(1)$ denotes the incoming probability that the jth edge takes the value 1.

8.10–4 MAP Decoding Using the Sum-Product Algorithm

A code $\mathcal{C}$ with codewords $\boldsymbol{c}_i$, $1 \leq i \leq M$, is used for communication over a memoryless channel. Codeword $\boldsymbol{c}$ is transmitted over the channel and $\boldsymbol{y}$ is received, and at the decoder we are interested in performing symbol-by-symbol maximum a posteriori decoding that maximizes $p(c_i \,|\, \boldsymbol{y})$. This can be written as

$$\begin{aligned}\hat{c}_i &= \arg\max_{1\leq m\leq M} p(c_{mi} \,|\, \boldsymbol{y}) \\ &= \arg\max_{1\leq m\leq M} \sum_{\sim c_{mi}} p(\boldsymbol{c}_m \,|\, \boldsymbol{y}) \\ &= \arg\max_{1\leq m\leq M} \sum_{\sim c_{mi}} p(\boldsymbol{c}_m) p(\boldsymbol{y} \,|\, \boldsymbol{c}_m) \\ &= \arg\max_{1\leq m\leq M} \sum_{\sim c_{mi}} p(\boldsymbol{c}_m) \prod_{i=1}^{n} p(y_i \,|\, c_{mi})\end{aligned} \tag{8.10–28}$$

This quantity has to be computed over all codewords $\boldsymbol{c}_m$.

For an arbitrary binary sequence of length n denoted by $\boldsymbol{c}$ we have

$$p(\boldsymbol{c}) = \begin{cases} \frac{1}{M} & \boldsymbol{c} \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases} \tag{8.10–29}$$

or equivalently we can write

$$p(\boldsymbol{c}) = \frac{1}{M}\delta[\boldsymbol{c} \in \mathcal{C}] \tag{8.10–30}$$

The MAP decoding rule then becomes

$$\hat{c}_i = \arg\max \sum_{\sim c_i} \delta[\boldsymbol{c} \in \mathcal{C}] \prod_{i=1}^{n} p(y_i \,|\, c_i) \tag{8.10–31}$$

[†] Or, equivalently, when the incoming two-dimensional message vector to each node is appropriately normalized such that the two components add to 1, i.e., if the messages are $\left(\frac{\mu(0)}{\mu(0)+\mu(1)}, \frac{\mu(1)}{\mu(0)+\mu(1)}\right)$.

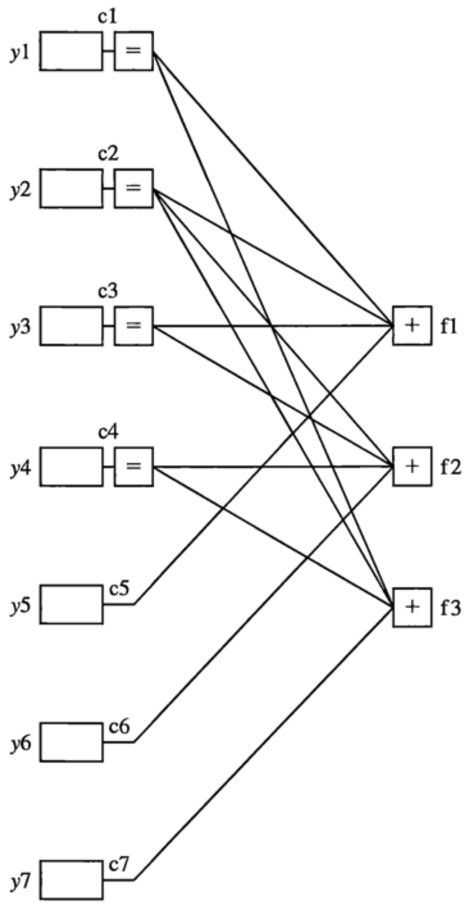


FIGURE 8.10–10
The code-channel factor graph for a (7, 4) Hamming code.

The factor $\delta[\boldsymbol{c} \in \mathcal{C}]$ determines the factor graph of the code, and factors $p(y_i|c_i)$ are nodes (functions) connected to the inputs (variable nodes) of the code factor graph with y_i as the input and $p(y_i|c_i)$ as the node function. The resulting factor graph for a (7, 4) Hamming code is shown in Figure 8.10–10. In this graph the leftmost squares represent the channel conditional probabilities $p(y_i|c_i)$.

The decoding process begins by supplying the channel outputs y_i as the variables to the variable nodes of the code-channel factor graph. Using the values of $p(y_i|c_i)$ and Equations 8.10–31 and 8.10–27, the decoder can apply the sum-product algorithm to find the marginal probabilities of each edge variable. The iterations are continued either for a fixed number of times or until a stopping criterion is satisfied. One such stopping criterion can be $\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0}$.

8.11
LOW DENSITY PARITY CHECK CODES

Low density parity check codes (LDPCs) are linear block codes that are characterized by a sparse parity check matrix. These codes were originally introduced in Gallager (1960, 1963), but were not widely studied for the next twenty years. Although Tanner

(1981) introduced the graphical representation of these codes, it was not until after the introduction of turbo codes and the iterative decoding algorithm that these codes were rediscovered by MacKay and Neal (1996) and MacKay (1999). Since then these codes have been the topic of active research in the coding community motivated by the excellent performance of these codes, which is realized by using iterative decoding schemes based on the sum-product algorithm. In fact, it has been shown that these codes are competitors to turbo codes in terms of performance and, if well designed, have better performance than turbo codes. Their excellent performance has resulted in their adoption in several communication and broadcasting standards.

Low density parity check codes are linear block codes with very large codeword length n usually in the thousands. The parity check matrix $\boldsymbol{H}$ for these codes is a large matrix with very few 1s in it. The term *low density* refers to the low density of 1s in the parity check matrix of these codes.

A *regular* low density parity check can be defined as a linear block code with a sparse $m \times n$ parity check matrix $\boldsymbol{H}$ satisfying the following properties.

1. There are w_r 1s in each row of $\boldsymbol{H}$, where $w_r \ll \min\{m, n\}$.
2. There are w_c 1s in each column of $\boldsymbol{H}$, where $w_c \ll \min\{m, n\}$.

The *density* of a low-density parity check code, denoted by r, is defined as the ratio of the total number of 1s in $\boldsymbol{H}$ to the total number of elements in $\boldsymbol{H}$. The density is given by

$$r = \frac{w_r}{n} = \frac{w_c}{m} \tag{8.11–1}$$

from which it is clear that

$$\frac{m}{n} = \frac{w_c}{w_r} \tag{8.11–2}$$

If the matrix $\boldsymbol{H}$ is full rank, then $m = n - k$

$$R_c = 1 - \frac{m}{n} = 1 - \frac{w_c}{w_r} \tag{8.11–3}$$

otherwise,

$$R_c = 1 - \frac{\operatorname{rank}(\boldsymbol{H})}{n} \tag{8.11–4}$$

The Tanner graph of a regular low density parity check code consists of the usual constraint and variable nodes. The low density constraint of the code, however, makes the degree of all constraint (parity-check) nodes equal to w_r which is much less than the code block length. Similarly the degree of all variable nodes is equal to w_c. The Tanner graph for an LDPC code is shown in Figure 8.11–1

The Tanner graph of LDPC codes usually is a graph with cycles. We have previously defined the girth of a graph as the length of the shortest cycle in that graph. Obviously a bipartite graph with cycles has a girth that is least equal to 4. A common decoding technique used for LDPC codes is the sum-product algorithm discussed in the preceding section. This algorithm is effective when the girth of the Tanner graph of the LDPC code is large. The reason for this behavior is that in order for the sum-product algorithm

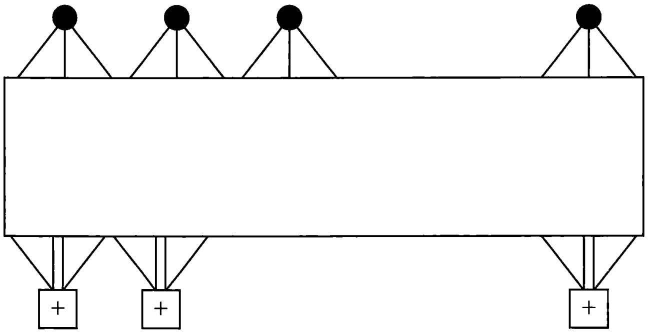


FIGURE 8.11–1
The Tanner graph for a regular LDPC code with $w_r = 4$ and $w_c = 3$.

to be effective on a graph with cycles, the value of the extrinsic information must be high. If the girth of the LDPC code is low, the information corresponding to a bit loops back to itself very soon, hence providing a small amount of extrinsic information and resulting in poor performance. Design techniques for LDPC codes with large girth are a topic of active research. We have seen in the preceding section that if the Tanner graph of a code has no cycles, then the sum-product algorithm converges in a finite number of steps. However, it has been shown that high-rate LDPC codes whose graph is cycle-free have low minimum distance and hence their bit error rate performance is poor.

An *irregular* LDPC code is one in which the number of 1s in rows and columns of $\boldsymbol{H}$ is low but is not constant for all rows and columns. Irregular low density parity check codes are usually described in terms of two *degree distribution polynomials* $\lambda(x)$ and $\rho(X)$, for variable nodes and constraint nodes, respectively. These polynomials are defined as

$$\begin{aligned} \lambda(x) &= \sum_{d=1}^{d_r} \lambda_d x^{d-1} \\ \rho(x) &= \sum_{d=1}^{d_c} \rho_d x^{d-1} \end{aligned} \tag{8.11–5}$$

where λ_d and ρ_d denote the fraction of all edges connected to variable and constraint nodes of degree d, respectively. It is clear that for a regular LDPC code we have

$$\begin{aligned} \lambda(x) &= x^{w_c-1} \\ \rho(x) &= x^{w_r-1} \end{aligned} \tag{8.11–6}$$

Very long irregular LDPC codes have been designed to operate within 0.0045 dB of the Shannon limit (see Chung et al. (2001)).

8.11–1 Decoding LDPC Codes

The two main algorithms used to decode LDPC codes are the bit-flipping algorithm and the sum-product algorithm, the latter also referred to as the *belief propagation algorithm.* The bit-flipping algorithm is a hard decision decoding algorithm with low complexity. The sum-product algorithm is a soft decision algorithm with higher complexity. We have already studied the sum-product algorithm in Section 8.10–3. Applying this algorithm to LDPC codes is straightforward and is based on applying Equations 8.10–31 and 8.10–27 to the code-channel factor graph.

The bit-flipping algorithm is a hard decision decoding algorithm. Let us assume that $\boldsymbol{y}$ is the hard channel output, i.e., the channel output quantized to 0 or 1. In the first step of the bit-flipping algorithm, the syndrome $\boldsymbol{s} = \boldsymbol{y}\boldsymbol{H}^t$ is computed. If the syndrome is zero, then we put $\hat{\boldsymbol{c}} = \boldsymbol{y}$ and stop. Otherwise, we consider the nonzero components of $\boldsymbol{s}$ corresponding to parity check equations that are not satisfied by the components of $\boldsymbol{y}$. The update of $\boldsymbol{y}$ is done by flipping those components of $\boldsymbol{y}$ that appear in the largest number of unsatisfied parity check equations. Equivalently, these are the node variables that are connected to the largest number of unsatisfied constraint nodes of the graph of the LDPC code. After the update the syndrome is computed again, and the whole process is repeated for a fixed number of iterations or until the syndrome is equal to zero. The interested reader can refer to Lin and Costello (2004) for more details on bit-flipping decoding and its various forms.

8.12

CODING FOR BANDWIDTH-CONSTRAINED CHANNELS — TRELLIS CODED MODULATION

In the treatment of block and convolutional codes, performance improvement was achieved by expanding the bandwidth of the transmitted signal by an amount equal to the reciprocal of the code rate. Recall for example that the improvement in performance achieved by an (n, k) binary block code with soft-decision decoding is approximately $10\log_{10}(R_c d_{\min} - k\ln 2/\gamma_b)$ compared with uncoded binary or quaternary PSK. For example, when $\gamma_b = 10$, the (24, 12) extended Golay code gives a coding gain of 5 dB. This coding gain is achieved at a cost of doubling the bandwidth of the transmitted signal and, of course, at the additional cost in receiver implementation complexity. Thus, coding provides an effective method for trading bandwidth and implementation complexity against transmitter power. This situation applies to digital communication systems that are designed to operate in the power-limited region where $R/W < 1$.

In this section, we consider the use of coded signals for bandwidth-constrained channels. For such channels, the digital communication system is designed to use bandwidth-efficient multilevel amplitude and phase modulation, such as PAM, PSK, DPSK, or QAM, and operates in the region where $R/W > 1$. When coding is applied to the bandwidth-constrained channel, a performance gain is desired without expanding the signal bandwidth. This goal can be achieved by increasing the number of signals over the corresponding uncoded system to compensate for the redundancy introduced by the code.

For example, suppose that a system employing uncoded four-phase PSK modulation achieves an $R/W = 2$ (bits/s)/Hz at an error probability of 10^{-6}. For this error rate the SNR per bit is $\gamma_b = 10.5$ dB. We may try to reduce the SNR per bit by use of coded signals, but this must be done without expanding the bandwidth. If we choose a rate $R_c = 2/3$ code, it must be accompanied by an increase in the number of signal points from four (2 bits per symbol) to eight (3 bits per symbol). Thus, the rate 2/3 code used in conjunction with eight-phase PSK, for example, yields the same data throughput as uncoded four-phase PSK. However, we recall that an increase in the number of signal phases from four to eight requires an additional 4 dB approximately in signal power to maintain the same error rate. Hence, if coding is to provide a benefit, the performance gain of the rate $2/3$ code must overcome this 4-dB penalty.

If the modulation is treated as a separate operation independent of the encoding, the use of very powerful codes (large-constraint-length convolutional codes or large-block-length block codes) is required to offset the loss and provide some significant coding gain. On the other hand, if the modulation is an integral part of the encoding process and is designed in conjunction with the code to increase the minimum Euclidean distance between pairs of coded signals, the loss from the expansion of the signal set is easily overcome and a significant coding gain is achieved with relatively simple codes. The key to this integrated modulation and coding approach is to devise an effective method for mapping the coded bits into signal points such that the minimum Euclidean distance is maximized. Such a method was developed by Ungerboeck (1982), based on the principle of *mapping by set partitioning*. We describe this principle by means of Examples 8.12–1 and 8.12–2.

Set partitioning We begin with a given signal constellation, such as M-ary PAM, or QAM or PSK, and partition the constellation into subsets in a way that the minimum Euclidean distance between signal points in a subset is increased with each partition. The following two examples illustrate the set partitioning method proposed by Ungerboeck.

EXAMPLE 8.12–1. AN 8-PSK SIGNAL CONSTELLATION. Let us partition the eight-phase signal constellation shown in Figure 8.12–1 into subsets of increasing minimum Euclidean distance. In the eight-phase signal set, the signal points are located on a circle of radius $\sqrt{\mathcal{E}}$ and have a minimum distance separation of

$$d_0 = 2\sqrt{\mathcal{E}} \sin \tfrac{1}{8}\pi = \sqrt{(2 - \sqrt{2})\mathcal{E}} = 0.765\sqrt{\mathcal{E}}$$

In the first partitioning, the eight points are subdivided into two subsets of four points each, such that the minimum distance between points increases to $d_1 = \sqrt{2\mathcal{E}}$. In the second level of partitioning, each of the two subsets is subdivided into two subsets of two points, such that the minimum distance increases to $d_2 = 2\sqrt{\mathcal{E}}$. This results in four subsets of two points each.

Finally, the last stage of partitioning leads to eight subsets, where each subset contains a single point. Note that each level of partitioning increases the minimum Euclidean distance between signal points. The results of these three stages of partitioning are illustrated in Figure 8.12–1. The way in which the coded bits are mapped into the partitioned signal points is described below.

EXAMPLE 8.12–2. A 16-QAM SIGNAL CONSTELLATION. The 16-point rectangular signal constellation shown in Figure 8.12–2 is first divided into two subsets by assigning alternate points to each subset as illustrated in the figure. Thus, the distance between points is increased from $2\sqrt{\mathcal{E}}$ to $2\sqrt{2\mathcal{E}}$ by the first partitioning. Further partitioning of the two subsets leads to greater separation in Euclidean distance between signal points as illustrated in Figure 8.12–2. It is interesting to note that for the rectangular signal

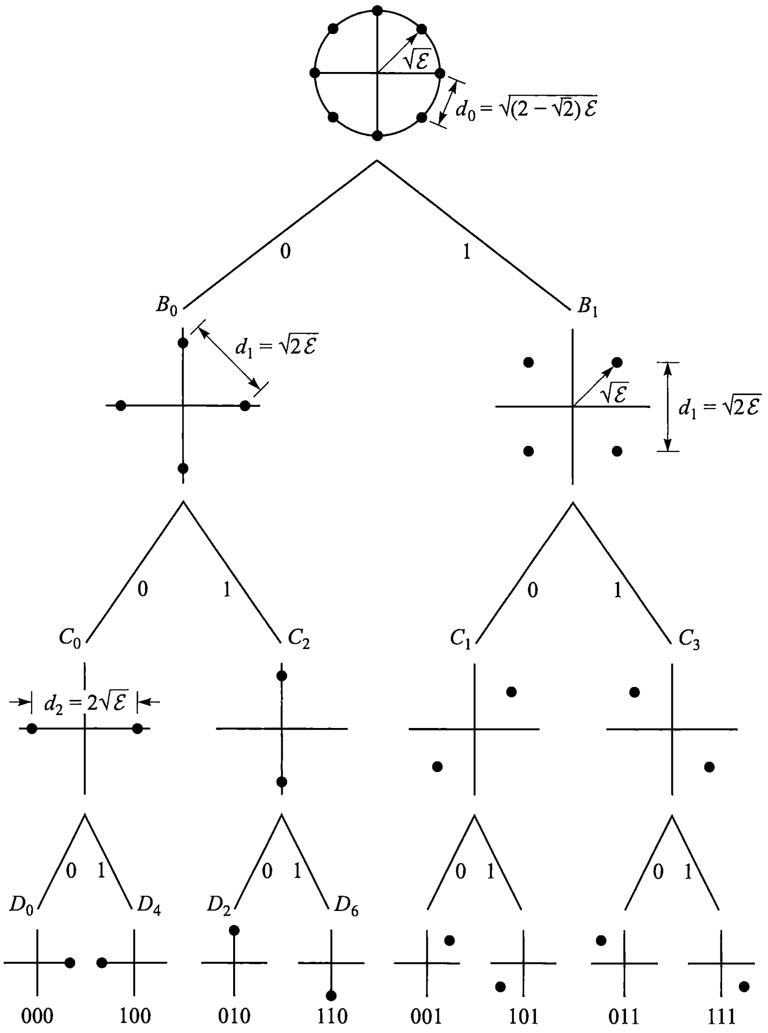

$\sqrt{\mathcal{E}}$
$d_0 = \sqrt{(2-\sqrt{2})\mathcal{E}}$
0
1
B_0
B_1
$d_1 = \sqrt{2\mathcal{E}}$
$\sqrt{\mathcal{E}}$
$d_1 = \sqrt{2\mathcal{E}}$
0
1
0
1
C_0
C_2
C_1
C_3
$d_2 = 2\sqrt{\mathcal{E}}$
0 1
0 1
0 1
0 1
D_0
D_4
D_2
D_6
000
100
010
110
001
101
011
111

FIGURE 8.12–1
Set partitioning of an 8-PSK signal set.

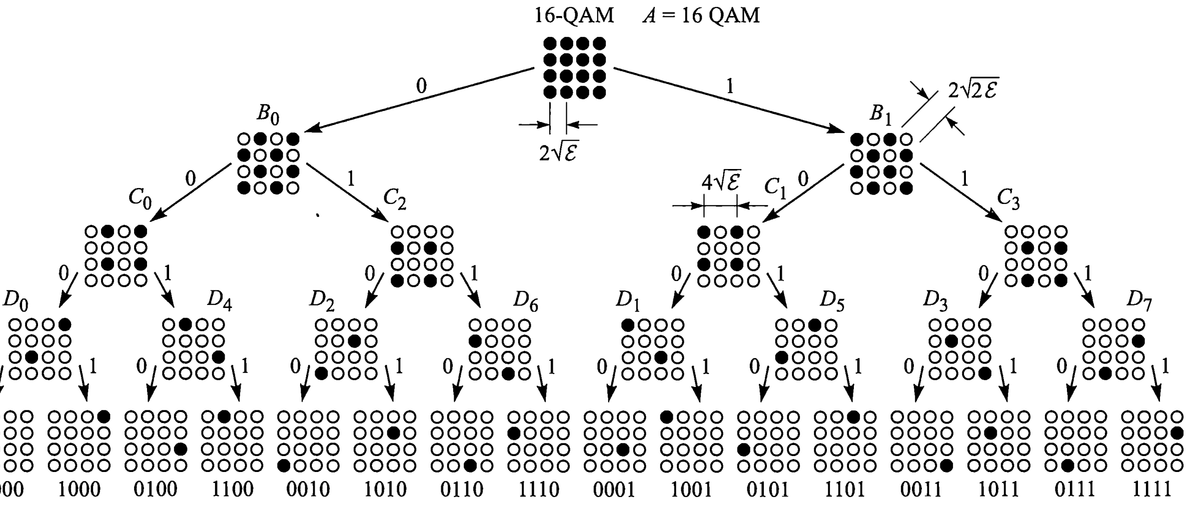

16-QAM
$A = 16$ QAM
0
1
$2\sqrt{\mathcal{E}}$
$2\sqrt{2\mathcal{E}}$
B_0
B_1
$4\sqrt{\mathcal{E}}$
0
1
0
1
C_0
C_2
C_1
C_3
D_0
D_4
D_2
D_6
D_1
D_5
D_3
D_7
0000
1000
0100
1100
0010
1010
0110
1110
0001
1001
0101
1101
0011
1011
0111
1111

FIGURE 8.12–2
Set partitioning of 16-QAM signal.

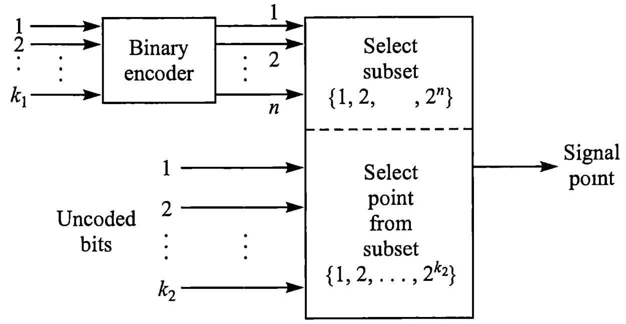


FIGURE 8.12–3
General structure of combined encoder/modulator.

constellations, each level of partitioning increases the minimum Euclidean distance by $\sqrt{2}$, i.e., $d_{i+1}/d_i = \sqrt{2}$ for all i.

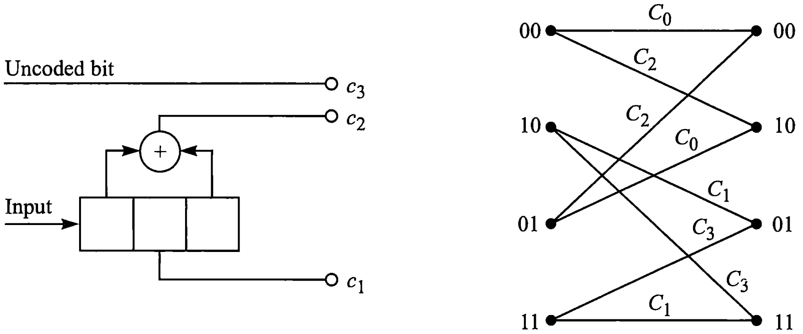
In these two examples, the partitioning was carried out to the limit where each subset contains only a single point. In general, this may not be necessary. For example, the 16-point QAM signal constellation may be partitioned only twice, to yield four subsets of four points each. Similarly, the eight-phase PSK signal constellation can be partitioned twice, to yield four subsets of two points each.

Trellis-coded modulation (TCM) The degree to which the signal is partitioned depends on the characteristics of the code. In general, the encoding process is performed as illustrated in Figure 8.12–3. A block of m information bits is separated into two groups of length k_1 and k_2, respectively. The k_1 bits are encoded into n bits, while the k_2 bits are left uncoded. Then, the n bits from the encoder are used to select one of the possible subsets in the partitioned signal set, while the k_2 bits are used to select one of 2^{k_2} signal points in each subset. When $k_2 = 0$, all m information bits are encoded.

The assignment of signal subsets to state transitions in the trellis is based on three heuristic rules devised by Ungerboeck (1982). The rules are

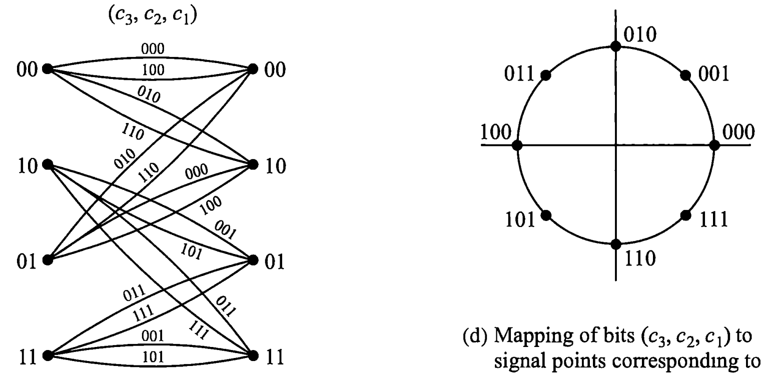
1. Use all subsets with equal frequency in the trellis.
2. Transitions originating from the same state or merging into the same state in the trellis are assigned subsets that are separated by the largest Euclidean distance.
3. Parallel state transitions (when they occur) are assigned signal points separated by the largest Euclidean distance. Parallel transitions in the trellis are characteristic of TCM that contains one or more uncoded information bits.

EXAMPLE 8.12–3. Consider the use of the rate 1/2 convolutional encoder shown in Figure 8.12–4a to encode one information bit while the second information bit is left uncoded. This code results in the four-state trellis shown in Figure 8.12–4b. When used in conjunction with an eight-point signal constellation, such as eight-point PSK or QAM, the two encoded output bits are used to select one of the four subsets in the partitioned signal constellation, while the remaining information bit is used to select one of the two points within each subset. Let us use the eight-point PSK consellation to complete this example. The four subsets assigned to the trellis in Figure 8.12–4b correspond to the subsets labeled C_0, C_1, C_2, C_3 in Figure 8.12–1. Note that the Euclidean distance of points within any subset is $d_2 = 2\sqrt{\mathcal{E}}$ and the largest minimum distance between signal points in any pair of subsets is $d_1 = \sqrt{2\mathcal{E}}$. The mappings of the coded bits (c_2, c_1) and the uncoded bit c_3 to the state transitions, using the convention (c_3, c_2, c_1) are shown in Figure 8.12–4c. We note that each trellis state has



(a) Encoder

(b) Four-state trellis



(c) Mapping of bits to state transitions

(d) Mapping of bits (c_3, c_2, c_1) to signal points corresponding to partition in Fig. 8.3-1 (note nonuniqueness of this mapping)

FIGURE 8.12–4
Four-state trellis-coded modulation with 8-PSK signal constellation.

two parallel transitions, corresponding to the two possible values of the uncoded bit. The phase assignments in the eight-point PSK constellation are shown in Figure 8.12–4d. It should be noted that the mapping of the bits (c_3, c_2, c_1) into the eight signal points in the constellation is not unique. Several other mappings are possible. For example, an equally good mapping is obtained if the four-point subsets B_0 and B_1 shown in Figure 8.12–1, are interchanged, so that the signal points in the subsets C_0, C_1, C_2, and C_3 will also change.

In general, the number of states $S = 2^{\nu}$ in the code trellis is a function of the number of memory elements in the encoder. Hence, we may increase the number of trellis states while maintaining the same code rate. For example, Figure 8.12–5 illustrates a rate 2/3 code that has eight trellis states. In this case, both information bits are coded.

Let us now evaluate the performance of the trellis-coded 8-PSK and compare its performance with that of uncoded 4-*PSK*, which we use as a reference in measuring the coding gain of the trellis-coded modulation. Uncoded 4-*PSK* employs the signal points in either subset B_0 or B_1 of Figure 8.12–1, for which the minimum distance of the signal points is $\sqrt{2\mathcal{E}}$. Note that the 4-PSK signal corresponds to a trivial one-state trellis with four parallel state transitions, as shown in Figure 8.12–6. The subsets D_0, D_2, D_4, and D_6 in Figure 8.12–1 are used as the signal points for the purpose of illustration.

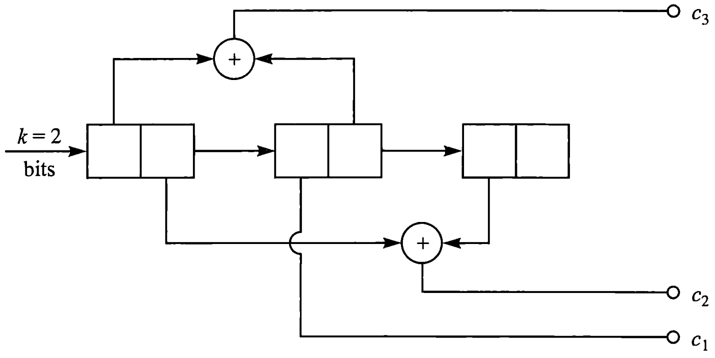
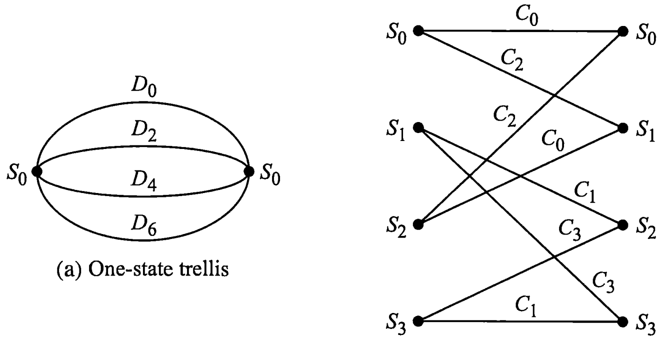

k = 2
bits
+
+
c_3
c_2
c_1

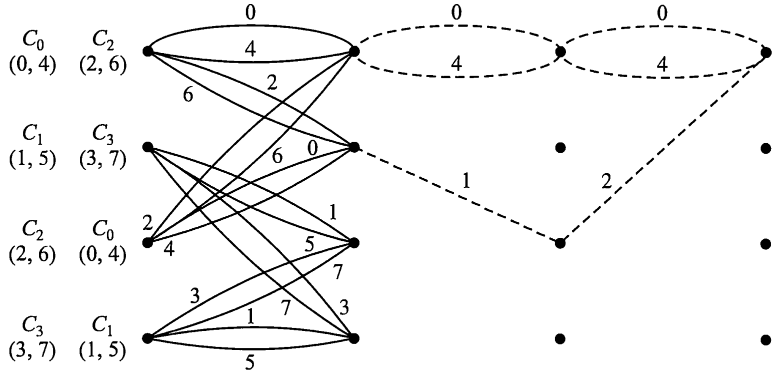
FIGURE 8.12–5
Rate $\frac{2}{3}$, eight-state trellis code.

For the trellis-coded 8-PSK modulation, we use the four-state trellis shown in Figure 8.12–4b and c. We observe that each branch in the trellis corresponds to one of the four subsets C_0, C_1, C_2, or C_3. As indicated above, for the eight-point constellation, each of the subsets C_0, C_1, C_2, and C_3 contains two signal points. Hence, the state transition C_0 contains the two signal points corresponding to the bits $(c_3c_2c_1) = (000)$ and (100), or (0, 4) in octal representation. Similarly, C_2 contains the two signal points corresponding to (010) and (110) or to (2, 6) in octal, C_1 contains the points corresponding to (001) and (101) or (1, 5) in octal, and C_3 contains the points corresponding


D_0
D_2
D_4
D_6
S_0
S_0
S_0
S_1
S_2
S_3
S_0
S_1
S_2
S_3
C_0
C_2
C_2
C_0
C_1
C_3
C_3
C_1

(a) One-state trellis

(b) Four-state trellis


C_0 (0, 4)
C_2 (2, 6)
C_1 (1, 5)
C_3 (3, 7)
C_2 (2, 6)
C_0 (0, 4)
C_3 (3, 7)
C_1 (1, 5)
0
4
6
2
6
0
2
4
1
5
7
3
1
7
3
5
0
4
0
4
1
2

(c) Four-state trellis

FIGURE 8.12–6
Uncoded 4-PSK and trellis-coded 8-PSK modulation.

to (011) and (111) or (3, 7) in octal. Thus, each transition in the four-state trellis contains two parallel paths, as previously indicated. As shown in Figure 8.12–6, any two signal paths that diverge from one state and remerge at the same state after more than one transition have a squared Euclidean distance of $d_0^2 + 2d_1^2 = d_0^2 + d_2^2$ between them. For example, the signal paths 0, 0, 0 and 2, 1, 2 are separated by $d_0^2 + d_2^2 = [(0.765)^2 + 4]\mathcal{E} = 4.585\mathcal{E}$. On the other hand, the squared Euclidean distance between parallel transitions is $d_2^2 = 4\mathcal{E}$. Hence, the minimum Euclidean distance separation between paths that diverge from any state and remerge at the same state in the four-state trellis is $d_2 = 2\sqrt{\mathcal{E}}$. The minimum distance in the trellis code is called the *free Euclidean distance* and denoted by D_{fed}.

In the four-state trellis of Figure 8.12–6b, $D_{\text{fed}} = 2\sqrt{\mathcal{E}}$. When compared with the Euclidean distance $d_0 = \sqrt{2\mathcal{E}}$ for the uncoded 4-PSK modulation, we observe that the four-state trellis code gives a coding gain of 3 dB.

We should emphasize that the four-state trellis code illustrated in Figure 8.12–6b is optimum in the sense that it provides the largest free Euclidean distance. Clearly, many other four-state trellis codes can be constructed, including the one shown in Figure 8.12–7, which consists of four distinct transitions from each state to all other states. However, neither this code nor any of the other possible four-state trellis codes gives a larger D_{fed}.

In the four-state trellis code, the parallel transitions were separated by the Euclidean distance $2\sqrt{\mathcal{E}}$, which is also D_{fed}. Hence, the coding gain of 3 dB is limited by the distance of the parallel transitions. Larger gains in performance relative to uncoded 4-PSK can be achieved by using trellis codes with more states, which allow for the elimination of the parallel transitions. Thus, trellis codes with eight or more states would use distinct transitions to obtain a larger D_{fed}.

For example, in Figure 8.12–8, we illustrate an eight-state trellis code due to Ungerboeck (1982) for the 8-PSK signal constellation. The state transitions for maximizing the free Euclidean distance were determined from application of the three basic rules given above. In this case, note that the minimum squared Euclidean distance is

$$D_{\text{fed}}^2 = d_0^2 + 2d_1^2 = 4.585\mathcal{E}$$

which, when compared with $d_0^2 = 2\mathcal{E}$ for uncoded 4-PSK, represents a gain of 3.6 dB. Ungerboeck (1982, 1987) has also found rate 2/3 trellis codes with 16, 32,

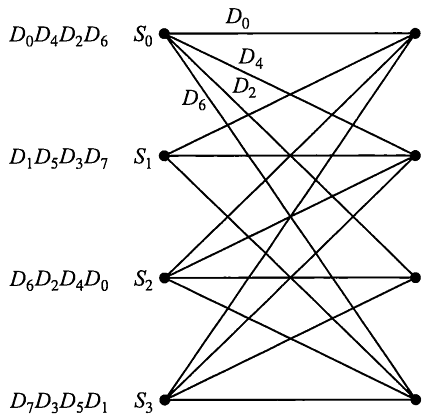


FIGURE 8.12–7
An alternative four-state trellis code.

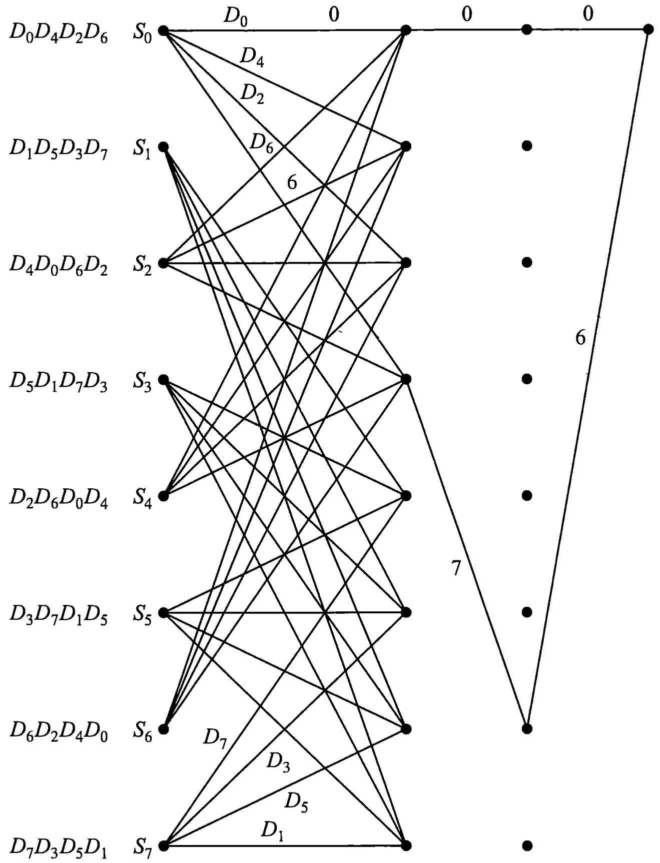


FIGURE 8.12–8
Eight-state trellis code for coded 8-PSK modulation.

64, 128, and 256 states that achieve coding gains ranging from 4 to 5.75 dB for 8-PSK modulation.

The basic principle of set partitioning is easily extended to larger PSK signal constellations that yield greater bandwidth efficiency. For example, 3 (bits/s)/Hz can be achieved with either uncoded 8-PSK or with trellis-coded 16-PSK modulation. Ungerboeck (1987) has devised trellis codes and has evaluated the coding gains achieved by simple rate $1/2$ and rate $2/3$ convolutional codes for the 16-PSK signal constellations. The results are summarized below.

Soft-decision Viterbi decoding for trellis-coded modulation is accomplished in two steps. Since each branch in the trellis corresponds to a signal subset, the first step in decoding is to determine the best signal point within each subset, i.e., the point in each subset that is closest in distance to the received point. We may call this *subset decoding*. In the second step, the signal point selected from each subset and its squared distance metric are used for the corresponding branch in the Viterbi algorithm to determine the signal path through the code trellis that has the minimum sum of squared distances from the sequence of received (noisy channel output) signals.

The error rate performance of the trellis-coded signals in the presence of additive Gaussian noise can be evaluated by following the procedure described in Section 8.2 for convolutional codes. Recall that this procedure involves the computation of the probability of error for all different error events and summing these error event probabilities to obtain a union bound on the first-event error probability. Note, however, that at high

SNR, the first-event error probability is dominated by the leading term, which has the minimum distance D_{fed}. Consequently, at high SNR, the first-event error probability is well approximated as

$$P_e \approx N_{\text{fed}} Q\left(\sqrt{\frac{D_{\text{fed}}^2}{2N_0}}\right) \tag{8.12–1}$$

where N_{fed} denotes the number of signal sequences with distance D_{fed} that diverge at any state and remerge at that state after one or more transitions.

In computing the coding gain achieved by trellis-coded modulation, we usually focus on the gain achieved by increasing D_{fed} and neglect the effect of N_{fed}. However, trellis codes with a large number of states may result in a large N_{fed} that cannot be ignored in assessing the overall coding gain.

In addition to the trellis-coded PSK modulations described above, powerful trellis codes have also been developed for PAM and QAM signal constellations. Of particular practical importance is the class of trellis-coded two-dimensional rectangular signal constellations. Figure 8.12–9 illustrates these signal constellations for M-QAM where $M = 16$, 32, 64, and 128. The $M = 32$ and 128 constellations have a cross pattern and are sometimes called *cross-constellations*. The underlying rectangular grid containing the signal points in M-QAM is called a *lattice of type* Z_2 (the subscript indicates the dimensionality of the space). When set partitioning is applied to this class of signal constellations, the minimum Euclidean distance between successive partitions is $d_{i+1}/d_i = \sqrt{2}$ for all i, as previously observed in Example 8.12–2.

Figure 8.12–10 illustrates an eight-state trellis code that can be used with any of the M-QAM rectangular signal constellations for which $M = 2^k$, where $k = 4, 5, 6, \ldots,$ etc. With the eight-state trellis, we associate eight signal subsets, so that any of the

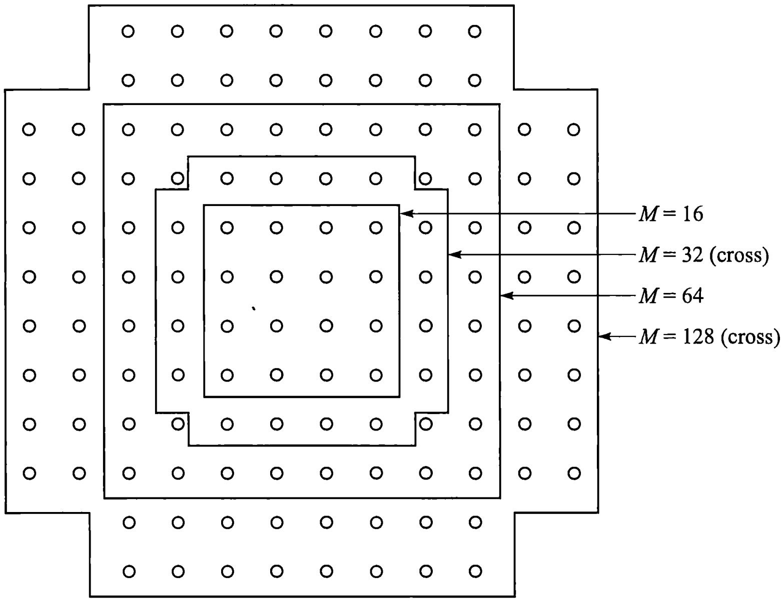


FIGURE 8.12–9
Rectangular two-dimensional (QAM) signal constellations.

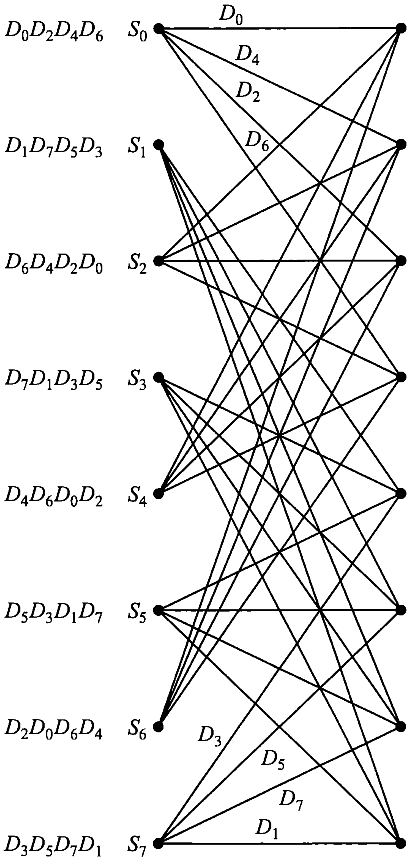


FIGURE 8.12–10
Eight-state trellis for rectangular QAM signal constellations.

M-QAM signal sets for $M \geq 16$ are suitable. For $M = 2^{m+1}$, two input bits ($k_1 = 2$) are encoded into $n = 3$ ($n = k_1 + 1$) bits that are used to select one of the eight subsets. The additional $k_2 = m - k_1$ input bits are used to select signal points within a subset, and result in parallel transitions in the eight-state trellis. Hence, 16-QAM with an 8-state trellis involves two parallel transitions in each branch of the trellis. More generally, the choice of an $M = 2^{m+1}$-point QAM signal constellation implies that the eight-state trellis contains 2^{m-2} parallel transitions in each branch.

The assignment of signal subsets to transitions is based on the same set of basic (heuristic) rules described above for the 8-PSK signal constellation. Thus, for the 8-state trellis, the four (branches) transitions originating from or leading to the same state are assigned either the subsets D_0, D_2, D_4, D_6 or D_1, D_3, D_5, D_7. Parallel transitions are assigned signal points contained within the corresponding subsets. This eight-state trellis code provides a coding gain of 4 dB. The Euclidean distance of parallel transitions exceeds the free Euclidean distance, and, hence, the code performance is not limited by parallel transitions.

Larger size trellis codes for M-QAM provide even larger coding gains. For example, trellis codes with 2^{ν} states for an $M = 2^{m+1}$ QAM signal constellation can be constructed by convolutionally encoding k_1 input bits into $k_1 + 1$ output bits. Thus, a rate $R_c = k_1/(k_1 + 1)$ convolutional code is employed for this purpose. Usually, the choice of $k_1 = 2$ provides a significant fraction of the total coding gain that is achievable. The additional $k_2 = m - k_1$ input bits are uncoded and are transmitted in each signal interval by selecting signal points within a subset.

TABLE 8.12–1
Coding Gains for Trellis-Coded PAM Signals

Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m = 1$ coding gain (dB) of 4-PAM versus uncoded 2-PAM	$m = 2$ coding gain (dB) of 8-PAM versus uncoded 4-PAM	$m \to \infty$ asymptotic coding gain (dB)	$m \to \infty$ N_{fed}
4	1	1/2	2.55	3.31	3.52	4
8	1	1/2	3.01	3.77	3.97	4
16	1	1/2	3.42	4.18	4.39	8
32	1	1/2	4.15	4.91	5.11	12
64	1	1/2	4.47	5.23	5.44	36
128	1	1/2	5.05	5.81	6.02	66

Source: Ungerboeck (1987).

Tables 8.12–1 to 8.12–3, taken from the paper by Ungerboeck (1987), provide a summary of coding gains achievable with trellis-coded modulation. Table 8.12–1 summarizes the coding gains achieved for trellis-coded (one-dimensional) PAM modulation with rate $1/2$ trellis codes. Note that the coding gain with a 128-state trellis code is 5.8 dB for octal PAM, which is close to the channel cutoff rate R_0 and less than 4 dB from the channel capacity limit for error rates in the range of 10^{-6}–10^{-8}. We should also observe that the number of paths N_{fed} with free Euclidean distance D_{fed} becomes large with an increase in the number of states.

Table 8.12–2 lists the coding gain for trellis-coded 16-PSK. Again, we observe that the coding gain for eight or more trellis states exceeds 4 dB, relative to uncoded 8-PSK. A simple rate $1/2$ code yields 5.33 dB gain with a 128-states trellis.

Table 8.12–3 contains the coding gains obtained with trellis-coded QAM signals. Relatively simple rate $2/3$ trellis codes yield a gain of 6 dB with 128 trellis states for $m = 3$ and 4.

The results in these tables clearly illustrate the significant coding gains that are achievable with relatively simple trellis codes. A 6-dB coding gain is close to the cutoff rate R_0 for the signal sets under consideration. Additional gains that would lead to

TABLE 8.12–2
Coding Gains for Trellis-Coded 16-PSK Modulation

Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m = 3$ coding gain (dB) of 16-PSK versus uncoded 8-PSK	$m \to \infty$ N_{fed}
4	1	1/2	3.54	4
8	1	1/2	4.01	4
16	1	1/2	4.44	8
32	1	1/2	5.13	8
64	1	1/2	5.33	2
128	1	1/2	5.33	2
256	2	2/3	5.51	8

Source: Ungerboeck (1987).

TABLE 8.12–3
Coding Gains for Trellis-Coded QAM Modulation

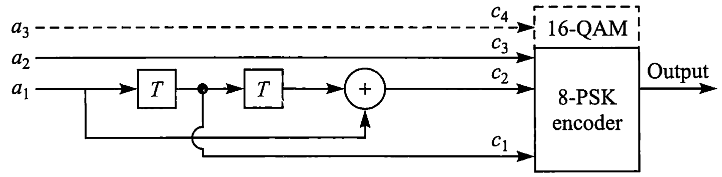
Number of states	k_1	Code rate $\frac{k_1}{k_1+1}$	$m=3$ gain (dB) of 16-QAM versus uncoded 8-QAM	$m=4$ gain (dB) of 32-QAM versus uncoded 16-QAM	$m=5$ gain (dB) of 64-QAM versus uncoded 32-QAM	$m=\infty$ asymptotic coding gain (dB)	N_{fed}
4	1	1/2	3.01	3.01	2.80	3.01	4
8	2	2/3	3.98	3.98	3.77	3.98	16
16	2	2/3	4.77	4.77	4.56	4.77	56
32	2	2/3	4.77	4.77	4.56	4.77	16
64	2	2/3	5.44	5.44	5.23	5.44	56
128	2	2/3	6.02	6.02	5.81	6.02	344
256	2	2/3	6.02	6.02	5.81	6.02	44

Source: Ungerboeck (1987).

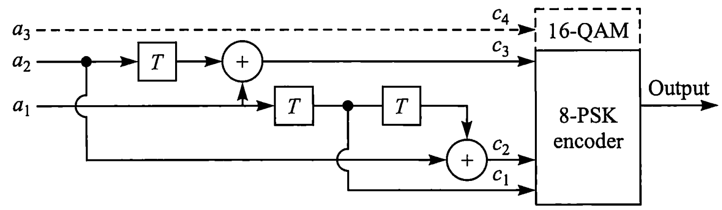
transmission in the vicinity of the channel capacity bound are difficult to attain without a significant increase in coding/decoding complexity. Continued partitioning of large signal sets quickly leads to signal point separation within any subset that exceeds the free Euclidean distance of the code. In such cases, parallel transitions are no longer the limiting factor on D_{fed}. Usually, a partition to eight subsets is sufficient to obtain a coding gain of 5–6 dB with simple rate $1/2$ or rate $2/3$ trellis codes with either 64 or 128 trellis states, as indicated in Tables 8.12–1 to 8.12–3.

Convolutional encoders for the linear trellis codes listed in Tables 8.12–1 to 8.12–3 for the M-PAM, M-PSK, and M-QAM signal constellations are given in the papers by Ungerboeck (1982, 1987). The encoders may be realized either with feedback or without feedback. For example Figure 8.12–11 illustrates three feedback-free convolutional encoders corresponding to 4-, 8-, and 16-state trellis codes for 8-PSK and 16-QAM signal constellations. Equivalent realizations of these trellis codes based on systematic convolutional encoders with feedback are shown in Figure 8.12–12. Usually, the systematic convolutional encoders are preferred in practical applications.

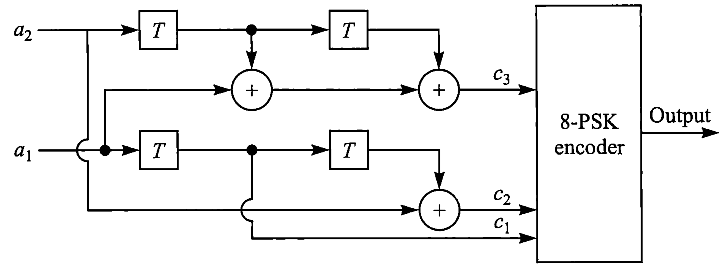
A potential problem with linear trellis codes is that the modulated signal sets are not usually invariant to phase rotations. This poses a problem in practical applications where differential encoding is usually employed to avoid phase ambiguities when a receiver must recover the carrier phase after a temporary loss of signal. For two-dimensional signal constellations, it is possible to achieve 180° phase invariance by use of a linear trellis code. However, it is not possible to achieve 90° phase invariance with a linear code. In such a case, a non-linear code must be used. The problem of phase invariance and differential encoding/decoding was solved by Wei (1984a,b), who devised linear and non-linear trellis codes that are rotationally invariant under either 180° or 90° phase rotations, respectively. For example, Figure 8.12–13 illustrates a non-linear eight-state convolutional encoder for a 32-QAM rectangular signal constellation that is invariant under 90° phase rotations. This trellis code has been adopted as an international standard (V.32 and V.33) for 9600 and 14,000 bits/s (high-speed) telephone line modems.



(a) 4-state encoder



(b) 8-state encoder



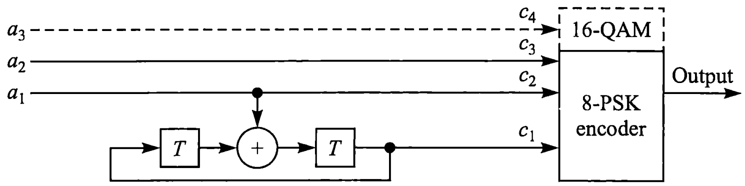
(c) 16-state encoder

FIGURE 8.12–11
Minimal feedback-free convolutional encoders for 8-PSK and 16-QAM signals. [*From Ungerboeck (1982). © 1982 IEEE.*]

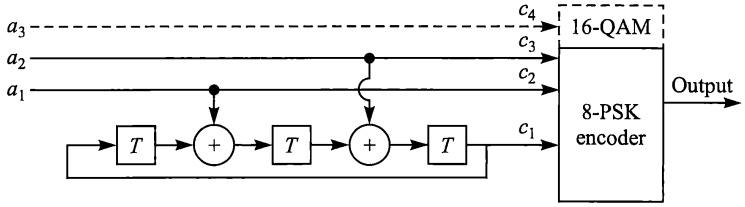
Trellis-coded modulation schemes have also been developed for multidimensional signals. In practical systems, multidimensional signals are transmitted as a sequence of either one-dimensional (PAM) or two-dimensional (QAM) signals. Trellis codes based on 4-, 8-, and 16-dimensional signal constellations have been constructed, and some of these codes have been implemented in commercially available modems. A potential advantage of trellis-coded multidimensional signals is that we can use smaller constituent two-dimensional signal constellations that allow for a trade-off between coding gain and implementation complexity. For example, a 16-state linear four-dimensional code, also designed by Wei (1987), is currently used as one of the codes for the V.34 telephone modem standard. The constituent two-dimensional signal constellation contains a maximum of 1664 signal points. The modem can transmit as many as 10 bits per symbol (eight uncoded bits) to achieve data rates as high as 33,600 bits/s. The papers by Wei (1987), Ungerboeck (1987), Gersho and Lawrence (1984), and Forney et al. (1984) treat multidimensional signal constellations for trellis-coded modulation.

8.12–1 Lattices and Trellis Coded Modulation

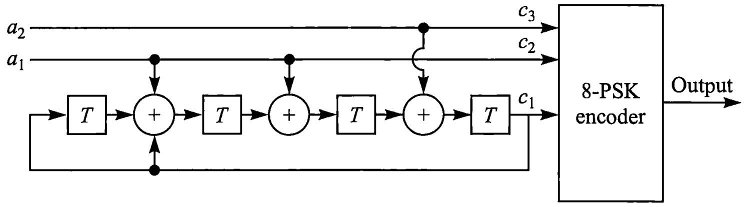
The set partitioning principles used in trellis coded modulation and the coding scheme based on set partitioning can be formulated in terms of lattices. We have defined lattices



(a) 4-state encoder



(b) 8-state encoder



(c) 16-state encoder

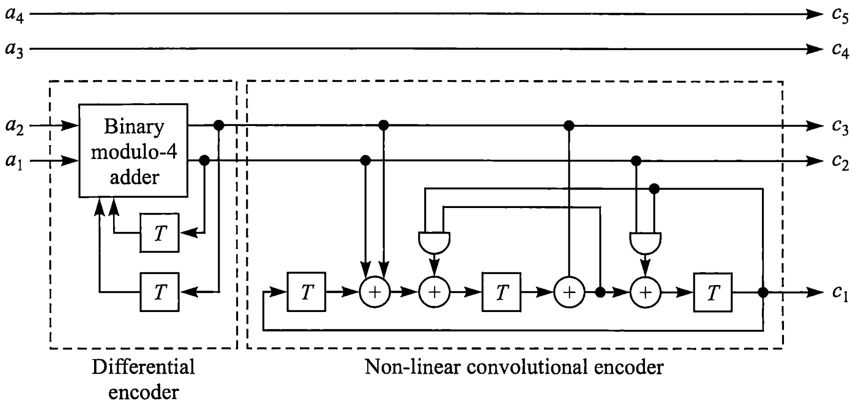
FIGURE 8.12–12
Equivalent realizations of systematic convolutional encoders with feedback for 8-PSK and 16-QAM. [*From Ungerboeck (1982).* © *1982 IEEE.*]

and sublattice in Section 4.7. If Λ' is a sublattice of lattice Λ and $\boldsymbol{c} \in \Lambda$ is arbitrary, we can define a shift of Λ' by $\boldsymbol{c}$, denoted by $\Lambda' + \boldsymbol{c}$ as the set of points of Λ' when each is shifted by $\boldsymbol{c}$. The result is called a *coset* of Λ' in Λ. If $\boldsymbol{c}$ is a member of Λ' then the coset is simply Λ'. The union of all distinct cosets of Λ' generate Λ, hence Λ can be partitioned into cosets where each coset is a shifted version of Λ'. The set of distinct cosets generated this way is denoted by Λ/Λ'. Each element of Λ/Λ' is a coset that can be represented by $\boldsymbol{c} \in \Lambda$; this element of the lattice is called the *coset representative*. The reader can compare this notion to the discussion of standard array and cosets in linear block codes discussed in Section 7.5 and notice the close relation. Coset representatives are similar to coset leaders. The set of coset representatives is represented by $[\Lambda/\Lambda']$, and the number of distinct cosets, called the *order of partition*, is denoted by $|\Lambda/\Lambda'|$. From this discussion we conclude that a lattice Λ can be partitioned into cosets and be written as the union of the cosets as

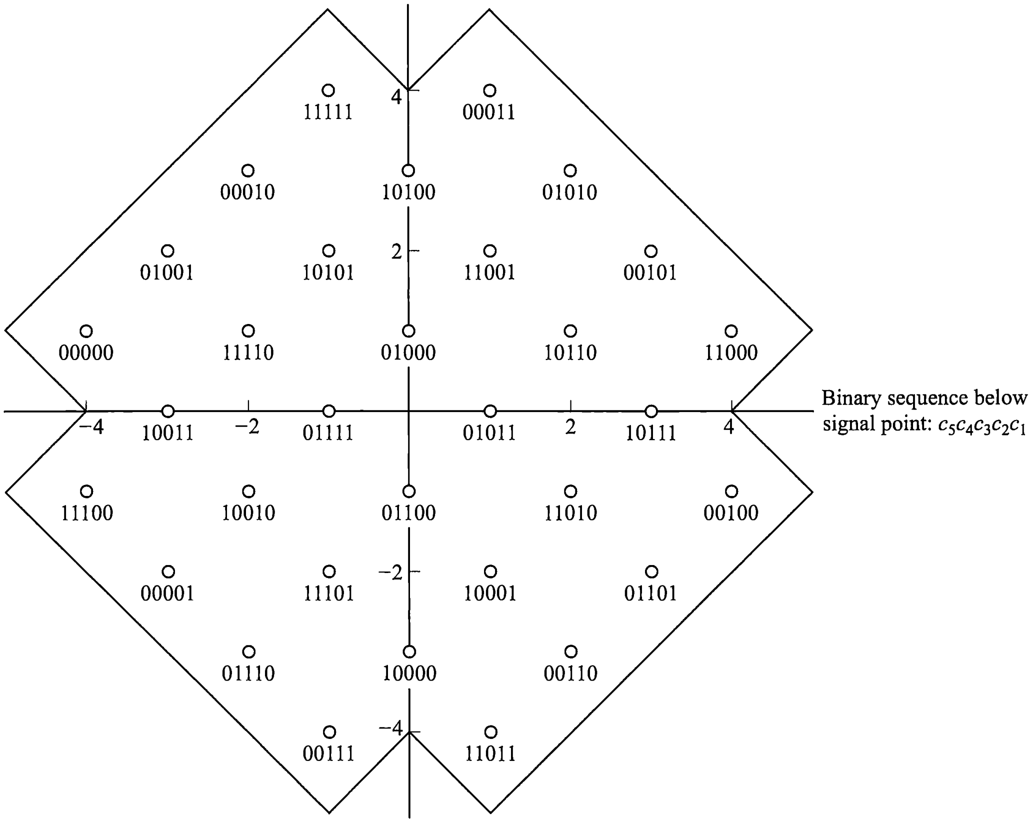
$$\Lambda = \bigcup_{i=1}^{L} \{\boldsymbol{c}_i + \Lambda'\} = [\Lambda/\Lambda'] + \Lambda' \tag{8.12–2}$$

where $L = |\Lambda/\Lambda'|$ is the partition order. This relation is called the coset decomposition of lattice Λ in terms of cosets of lattice Λ'.

The set partitioning of a constellation can be compared with the coset decomposition of a lattice. Let us assume a lattice Λ is decomposed using sublattice Λ' such that


a_4
c_5
a_3
c_4
a_2
a_1
Binary modulo-4 adder
T
T
c_3
c_2
c_1
Differential encoder
Non-linear convolutional encoder

(a) Encoder


11111
00011
00010
10100
01010
01001
10101
11001
00101
00000
11110
01000
10110
11000
10011
01111
01011
10111
11100
10010
01100
11010
00100
00001
11101
10001
01101
01110
10000
00110
00111
11011
4
2
−2
−4
Binary sequence below signal point: $c_5c_4c_3c_2c_1$

(b) 32-point QAM (cross) signal

FIGURE 8.12–13
Eight-state non-linear convolutional encoder for 32-QAM signal set that exhibits invariance under 90° phase rotations.

the order of the partition $|\Lambda/\Lambda'|$ is equal to 2^n, then each coset can serve as one of the partitions used in Ungerboeck's set partitioning. An (n, k_1) code is used to encode k_1 information bits into a binary sequence of length n which select one of the 2^n cosets in the lattice decomposition. The k_2 uncoded bits are used to select a point in the coset. Note that the number of elements in a coset is equal to the number of elements of the sublattice Λ' which is infinite, selection of a point in the coset determines the signal

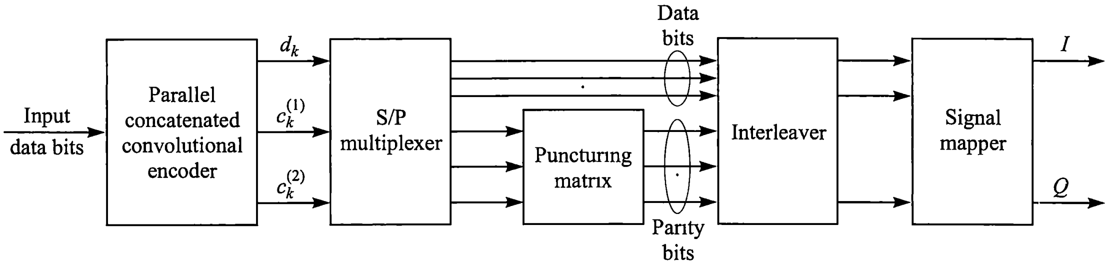

Input data bits
Parallel concatenated convolutional encoder
d_k
$c_k^{(1)}$
$c_k^{(2)}$
S/P multiplexer
Data bits
Puncturing matrix
Parity bits
Interleaver
Signal mapper
I
Q

FIGURE 8.12–14
Encoder for concatenation of a PCCC (turbo code) with TCM.

space boundary, thus determining the *shaping*. The total coding gain can then be defined as the product of two factors, *the fundamental coding gain* and the *shaping gain*. The shaping gain measures the amount of power reduction resulting from using a close to spherically shaped boundary and is independent from the convolutional code and the lattice used. The value of the shaping gain is limited to 1.53 dB as was discussed in Section 4.7. The interested reader is referred to Forney (1988).

8.12–2 Turbo-Coded Bandwidth Efficient Modulation

The performance of TCM can be further improved by code concatenation. There are several different methods described in the literature. We shall briefly describe two schemes for code concatenation using parallel concatenated codes, which we simply refer to as turbo coding.

In one scheme, described in the paper by Le Goff et al. (1994), the information sequence is fed to a binary turbo encoder that employs a parallel concatenation of a component convolutional code with interleaving to generate a systematic binary turbo code. As shown in Figure 8.12–14, the output of the turbo encoder is ultimately connected to the signal mapper after the binary sequence from the turbo code has been appropriately multiplexed, the parity bit sequence has been punctured to achieve the desired code rate, and the data and parity sequences have been interleaved. Gray mapping is typically used in mapping coded bits to modulation signal points, separately for the in-phase (I) and quadrature (Q) signal components.

Figure 8.12–15 illustrates the block diagram of the decoder for this turbo coding scheme. Based on each received I and Q symbol, the receiver computes the logarithm of the likelihood ratio or the MAP of each systematic bit and each parity bit.

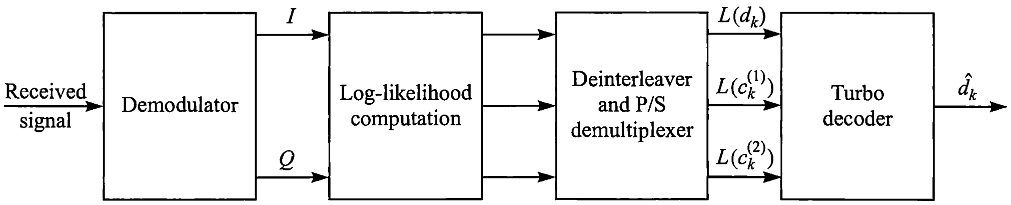

Received signal
Demodulator
I
Q
Log-likelihood computation
Deinterleaver and P/S demultiplexer
$L(d_k)$
$L(c_k^{(1)})$
$L(c_k^{(2)})$
Turbo decoder
$\hat{d}_k$

FIGURE 8.12–15
Decoder for concatenated PCCC/TCM code.

After deinterleaving, depuncturing, and demultiplexing of these logarithmic metrics, the systematic and parity bit information are fed to the standard binary turbo decoder.

This scheme for constructing turbo-coded bandwidth efficient modulation imposes no constraints on the type or size of the signal constellation. In addition, this scheme can be matched to any conventional binary turbo code. In fact, this scheme is also suitable if the turbo code is replaced by a serially concatenated convolutional code.

A second scheme employs a conventional Ungerboeck trellis code with interleaving to yield a parallel concatenated TCM. The basic configuration of the turbo TCM encoder, as described in the paper by Robertson and Wörz (1998), is illustrated in Figure 8.12–16. To avoid a rate loss, the parity sequence is punctured, as described below, in such a way that all information bits are transmitted only once, and the parity bits from the two encoders are alternately punctured. The block interleaver operates on groups of $m - 1$ information bits, where the signal constellation consists of 2^m signal points.

To illustrate the group interleaving and puncturing, let us consider a rate $R_c = \frac{2}{3}$ TCM code, a block interleaver of length $N = 6$, and 8-PSK modulation ($m = 3$). Hence, the number of information bits per block is $N(m-1) = 12$, and the interleaving is performed on pairs of information bits as shown in Figure 8.12–16 where, for example, a pair of bits in an even position (2, 4, 6) is mapped to another even position and a pair of bits in an odd position is mapped to another odd position. The output of the second

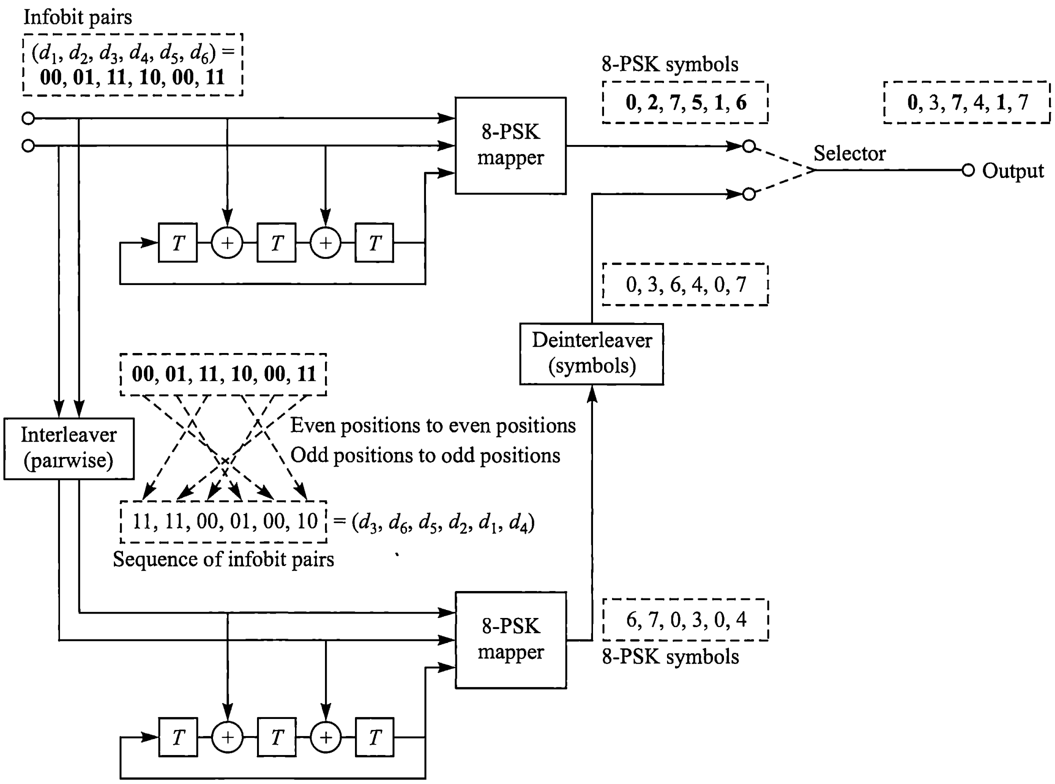

Infobit pairs
$(d_1, d_2, d_3, d_4, d_5, d_6) =$
00, 01, 11, 10, 00, 11
8-PSK symbols
0, 2, 7, 5, 1, 6
0, 3, **7**, 4, **1**, 7
8-PSK mapper
Selector
Output
T
+
0, 3, 6, 4, 0, 7
Deinterleaver (symbols)
00, 01, 11, 10, 00, 11
Even positions to even positions
Odd positions to odd positions
Interleaver (pairwise)
11, 11, 00, 01, 00, 10 = $(d_3, d_6, d_5, d_2, d_1, d_4)$
Sequence of infobit pairs
6, 7, 0, 3, 0, 4
8-PSK symbols

FIGURE 8.12–16
Turbo TCM encoder shown for 8-PSK with two-dimensional component codes of memory 3. An example of interleaving with $N = 6$ is shown. Bold letters indicate that symbols or pairs of bits correspond to the upper encoder. [*From Robertson and Wörz (1998); © 1998 IEEE.*]

TCM encoder is deinterleaved symbol-wise as illustrated in Figure 8.12–16, and the output symbol sequence is obtained by puncturing the two signal-point sequences, i.e., by selecting every other symbol from each of the two sequences. That is, we select the even-numbered symbols from the top symbol mapper and the odd-numbered symbols from the bottom symbol mapper. (In general, some of the information bits can remain uncoded, depending on the signal constellation and the signal mapping. In this example, both information bits are coded.)

A block diagram of the turbo decoder is shown in Figure 8.12–17. In the conventional binary iterative turbo decoder, each output of each component decoder is usually split into three parts, namely, the systematic part, the a priori part, and the extrinsic part, where only the latter is passed between the two decoders. In this TCM scheme, the systematic part cannot be separated from the extrinsic component, because the noise that affects the parity component also affects the systematic component due to the fact that both components are transmitted by the same symbol. This implies that the output of the decoders can be split into only two components, namely, the a priori information and the extrinsic-systematic information. Hence, each decoder passes the extrinsic-systematic information to the other decoder. Each decoder ignores those symbols where the pertinent parity bit was not sent and obtains the systematic information

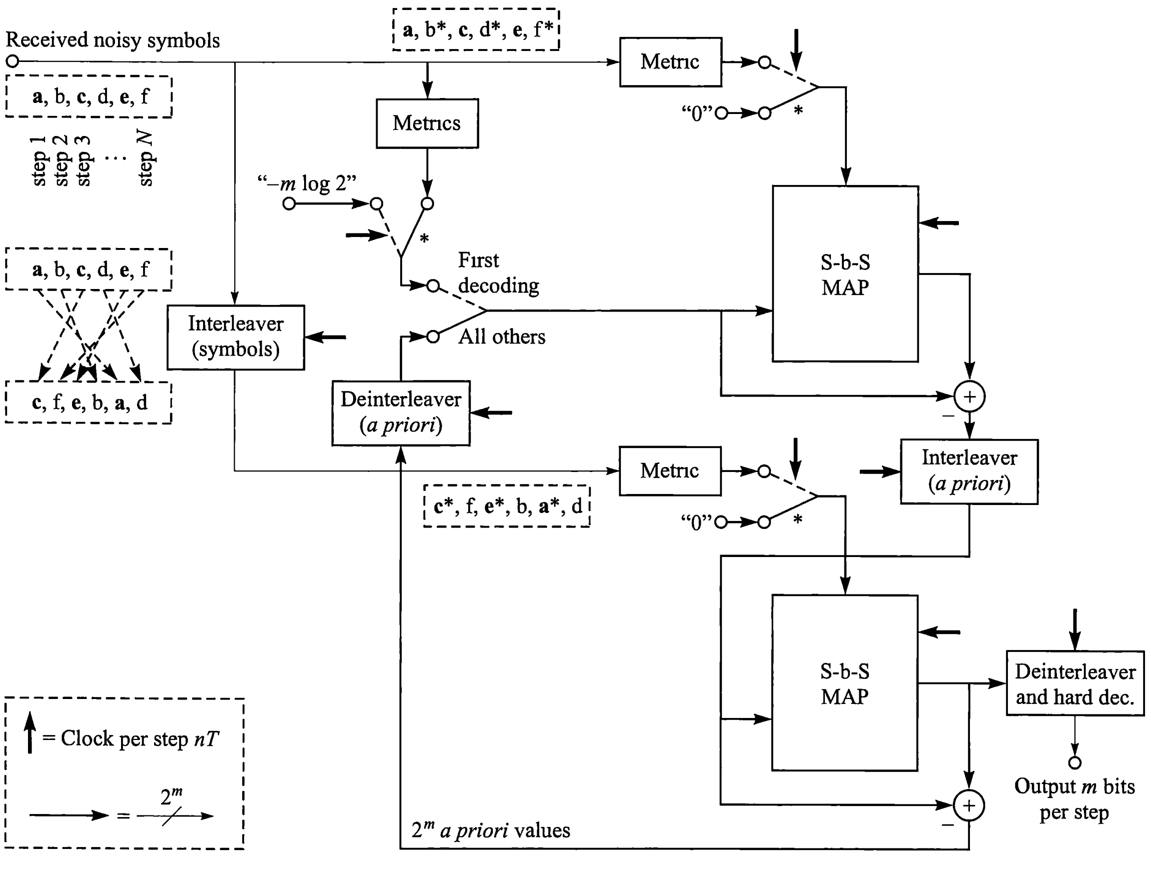

Received noisy symbols
a, b, c, d, e, f
step 1
step 2
step 3
…
step N
a, b, c, d, e, f
c, f, e, b, a, d
= Clock per step nT
2^m
a, b*, c, d*, e, f*
Metrics
"$-m$ log 2"
*
First decoding
All others
Interleaver (symbols)
Deinterleaver (a $priori$)
Metric
"0"
*
S-b-S MAP
+
−
Interleaver (a $priori$)
Metric
c*, f, e*, b, a*, d
"0"
*
S-b-S MAP
Deinterleaver and hard dec.
Output m bits per step
+
−
2^m a $priori$ values

FIGURE 8.12–17

Turbo TCM decoder corresponding to the encoder in Figure 8.12–16. [*From Robertson and Wörz (1998); © 1998 IEEE.*]

through its a priori input. In the first iteration, the a priori input of the first decoder is initialized with the missing systematic information. Details of the iterative decoder computations are given in the paper by Robertson and Wörz (1998). An additional coding gain of about 1.7 dB has been achieved by use of a turbo TCM compared to conventional TCM, at error rates in the vicinity of 10^{-4}. This means that turbo TCM achieves a performance close to the Shannon capacity on an AWGN channel.

8.13
BIBLIOGRAPHICAL NOTES AND REFERENCES

In parallel with the developments on block codes are the developments in convolutional codes, which were invented by Elias (1955). The major problem in convolutional coding was decoding. Wozencraft and Reiffen (1961) described a sequential decoding algorithm for convolutional codes. This algorithm was later modified and refined by Fano (1963), and it is now called the *Fano algorithm*. Subsequently, the stack algorithm was devised by Zigangirov (1966) and Jelinek (1969), and the Viterbi algorithm was devised by Viterbi (1967). The optimality and the relatively modest complexity for small constraint lengths have served to make the Viterbi algorithm the most popular in decoding of convolutional codes with $K \leq 10$.

One of the most important contributions in coding during the 1970s was the work of Ungerboeck and Csajka (1976) on coding for bandwidth-constrained channels. In this paper, it was demonstrated that a significant coding gain can be achieved through the introduction of redundancy in a bandwidth-constrained channel, and trellis codes were described for achieving coding gains of 3–4 dB. This work has generated much interest among researchers and has led to a large number of publications over the past 15 years. A number of references can be found in the papers by Ungerboeck (1982, 1987) and Forney et al. (1984). The papers by Benedetto et al. (1988, 1994) focus on applications and performance evaluation. Additional papers on coded modulation for bandwidth-constrained channels may also be found in the Special Issue on Voiceband Telephone Data Transmission, *IEEE Journal on Selected Areas in Communication* (September 1984, August 1989, and December 1989). A comprehensive treatment of trellis-coded modulation is given in the book by Biglieri et al. (1991).

A major new advance in coding and decoding is the construction of parallel and serially concatenated codes with interleaving, and the decoding of such codes using iterative MAP algorithms. Both PCCC and SCCC have been shown to yield performance very close to the Shannon limit with iterative decoding. PCCCs, called turbo codes, and the use of iterative decoding were first described in a paper by Berrou et al. (1993). Serially concatenated codes with interleaving and their performance have been treated in the paper by Benedetto et al. (1998). Turbo coding and decoding is also treated in the books by Heegard and Wicker (1999), Johannesson and Zigangirov (1999), and Schlegel (1997). Performance bounds for turbo codes are given in the paper by Duman and Salehi (1997) and Sason and Shamai (2001a, b).

Low density parity check codes were introduced by the pioneering work of Gallager (1963). Tanner (1981) studied the relation between these codes and graphs, and the work of MacKay and Neal (1996) reinstated the interest in these works. Wiberg et al. (1995), Wiberg (1996), and Forney (2000) extended the work of Tanner on the relation between codes and graphs.

In addition to the references given above on coding, decoding, and coded signal design, we should mention the collection of papers published by the IEEE Press entitled *Key Papers in the Development of Coding Theory*, edited by Berlekamp (1974). This book contains important papers that were published in the first 25 years of coding theory. We should also cite the Special Issue on Error-Correcting Codes, *IEEE Transactions on Communications* (October 1971). Finally, the survey papers by Calderbank (1998), Costello et al. (1998), and Forney and Ungerboeck (1998) highlight the major developments in coding and decoding over the past 50 years and include a large number of references.

PROBLEMS

8.1 A convolutional code is described by

$$g_1 = [101], \quad g_2 = [111], \quad g_3 = [111]$$

1. Draw the encoder corresponding to this code.
2. Draw the state-transition diagram for this code.
3. Draw the trellis diagram for this code.
4. Find the transfer function and the free distance of this code.
5. Verify whether or not this code is catastrophic.

8.2 The convolutional code of Problem 8.1 is used for transmission over an AWGN channel with hard decision decoding. The output of the demodulator detector is $(101001011110111\cdots)$. Using the Viterbi algorithm, find the transmitted sequence, assuming that the convolutional code is terminated at the zero state.

8.3 Repeat Problem 8.1 for a code with

$$g_1 = [110], \quad g_2 = [101], \quad g_3 = [111]$$

8.4 The block diagram of a binary convolutional code is shown in Figure P8.4.
1. Draw the state diagram for the code.
2. Find the transfer function of the code $T(Z)$.
3. What is d_{free}, the minimum free distance of the code?

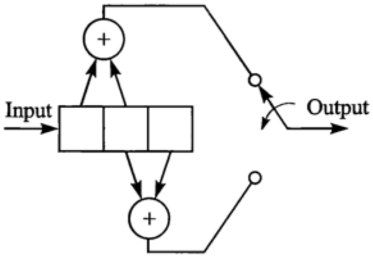


FIGURE P8.4

4. Assume that a message has been encoded by this code and transmitted over a binary symmetric channel with an error probability of $p = 10^{-5}$. If the received sequence is

$$r = (110, 110, 110, 111, 010, 101, 101)$$

using the Viterbi algorithm, find the most likely information sequence, assuming that the convolutional code is terminated at the zero state.
5. Find an upper bound to the bit error probability of the code when the above binary symmetric channel is employed. Make any reasonable approximation.

8.5 The block diagram of a (3, 1) convolutional code is shown in Figure P8.5.
1. Draw the state diagram of the code.
2. Find the transfer function $T(Z)$ of the code.
3. Find the minimum free distance (d_{free}) of the code, and show the corresponding path (at distance d_{free} from the all-zero codeword) in the trellis.
4. Determine $\boldsymbol{G}(D)$ for this code. Use $\boldsymbol{G}(D)$ to determine whether this code is catastrophic.
5. Determine $\boldsymbol{G}(D)$ for the RSCC equivalent to this code, and sketch a block diagram of it.
6. Assume that four information bits (x_1, x_2, x_3, x_4), followed by two zero bits have been encoded and sent via a binary-symmetric channel with crossover probability equal to 0.1. The received sequence is (111, 111, 111, 111, 111, 111). Use the Viterbi decoding algorithm to find the most likely data sequence, assuming that the convolutional code is terminated at the zero state.

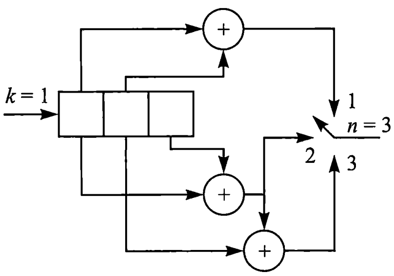


FIGURE P8.5

8.6 In the convolutional code generated by the encoder shown in Figure P8.6:
1. Find the transfer function of the code in the form $T(Y, Z)$.
2. Find d_{free} of the code.
3. If the code is used on a channel with hard decision Viterbi decoding, assuming the crossover probability of the channel is $p = 10^{-6}$, use the hard decision bound to find an upper bound on the average bit error probability of the code.

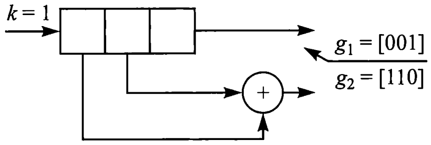


FIGURE P8.6

8.7 Figure P8.7 depicts a rate 1/2, constraint length $K = 2$, convolutional code.
1. Sketch the tree diagram, the trellis diagram, and the state diagram.
2. Solve for the transfer function $T(Y, Z, J)$, and from this, specify the minimum free distance.

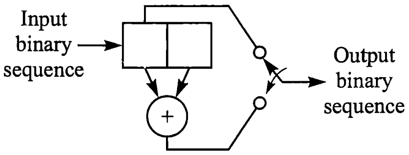


FIGURE P8.7

8.8 A rate $1/2$, $K = 3$, binary convolutional encoder is shown in Figure P8.8.

1. Draw the tree diagram, the trellis diagram, and the state diagram.
2. Determine the transfer function $T(Y, Z, J)$, and from this, specify the minimum free distance.
3. Determine the RSCC equivalent to this code, and sketch a block diagram of it.
4. Determine whether this code is catastrophic.

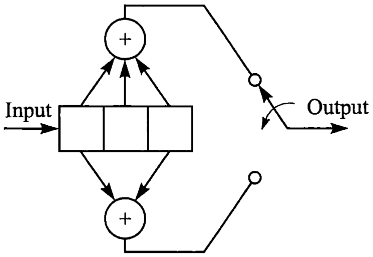


FIGURE P8.8

8.9 A $k = 1$, $K = 3$, and $n = 2$ convolutional code is characterized by $g_1 = [001]$ and $g_2 = [101]$.

1. Draw the state diagram for the encoder.
2. Determine the transfer function of the code in the form $T(Y, Z)$.
3. Is this code a catastrophic code? Why?
4. Determine the free distance of the code.
5. If the code is used with hard decision decoding on a channel with crossover probability of $p = 10^{-3}$, determine an upper bound on the average *bit error probability* of the code.

8.10 The block diagram for a convolutional code is given in Figure P8.10.

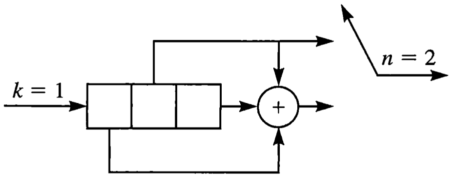


FIGURE P8.10

1. Draw the state transition diagram for this code.
2. Is this code catastrophic? Why?
3. What is the transfer function for this code?
4. What is the free distance of this code?
5. Assuming that this code is used for binary data transmission over a binary symmetric channel with crossover probability of 10^{-3}, find a bound on the resulting bit error probability.

8.11 The convolutional code shown in Figure P8.10 is used with a binary antipodal signaling scheme for transmission over an additive noise channel with input-output relation

$$r_i = c_i + n_i$$

where $c_i \in \{\pm\sqrt{\mathcal{E}_c}\}$ and noise components are iid random variables with PDF

$$p(n) = \frac{1}{2}e^{-|n|}$$

The receiver uses a soft decision ML decoding scheme.

1. Show that the optimal decoding rule is given by

$$\boldsymbol{c}^{(m)} = \min_{\boldsymbol{c}\in\mathcal{T}} \sum_j \left|\boldsymbol{r}_j - \boldsymbol{c}_j\right|$$

2. Find an upper bound for the average bit error probability for this system. Is this a useful bound? Why?
3. Assuming that $\mathcal{E}_c = 1$ and the code is terminated at the zero state, determine the most likely information sequence if the received output of the matched filter is

$$\boldsymbol{r} = (-1, -1, 1.5, 2, 0.7, -0.5, -0.8, -3, 3, 0.2, 0, 1)$$

4. If in part 3 instead of soft decision decoding, hard decision is employed, what is the most likely information sequence?
5. Answer part 2 for hard decision decoding.

8.12 The block diagram for a convolutional encoder is shown in Figure P8.12.

1. What is the number of states for this code?
2. Determine the transfer function $T(Y, Z)$ for this code, and find its free distance.
3. How many paths at the free distance exist in this code?
4. Is this code catastrophic? Why?
5. Assuming that this code is used for transmission over a binary symmetric channel with a crossover probability of 10^{-4}, find a bound on the bit error probability.

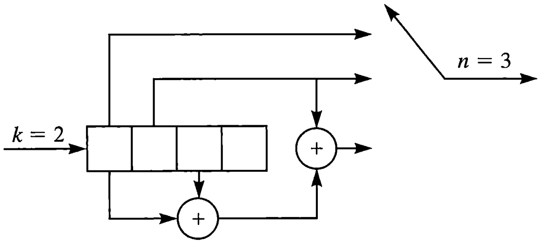


FIGURE P8.12

8.13 For the convolutional code shown in Figure P8.12:

1. Determine the matrix $\boldsymbol{G}(D)$.
2. Determine the encoded sequence for the input sequence $\boldsymbol{u} = (1001111001)$ using $\boldsymbol{G}(D)$ found in part 1.
3. Directly determine the encoded sequence corresponding to $\boldsymbol{u}$ given in part 2, and compare it with the sequence obtained using $\boldsymbol{G}(D)$.
4. Using $\boldsymbol{G}(D)$, determine whether this code is catastrophic.

8.14 A $k = 1$, $K = 3$, and $n = 2$ convolutional code is characterized by $\boldsymbol{g}_1 = [001]$ and $\boldsymbol{g}_2 = [110]$.

1. Find the transfer function of the code in the form $T(Y, Z)$.
2. Is this code catastrophic? Why?
3. Find d_{free} for the code.

4. If the code is used on an AWGN channel using BPSK with hard decision Viterbi decoding, assuming $\mathcal{E}_b/N_0 = 12.6$ dB, find an upper bound on the average bit error probability of the code.

8.15 Use Tables 8.3–1 to 8.3–11 to sketch the convolutional encoders for the following codes:

1. Rate 1/2, $K = 5$, maximum free distance code
2. Rate 1/3, $K = 5$, maximum free distance code
3. Rate 2/3, $K = 2$, maximum free distance code

8.16 Draw the state diagram for the rate 2/3, $K = 2$, convolutional code indicated in Problem 8.15, part 3, and, for each transition, show the output sequence and the distance of the output sequence from the all-zero sequence.

8.17 Consider the $K = 3$, rate 1/2, convolutional code shown in Figure P8.17. Suppose that the code is used on a binary symmetric channel and the received sequence for the first eight branches is 0001100000001001. Trace the decisions on a trellis diagram, and label the survivors' Hamming distance metric at each node level. If a tie occurs in the metrics required for a decision, always choose the upper path (arbitrary choice).

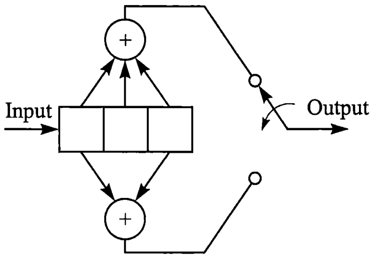


FIGURE P8.17

8.18 Use the transfer function derived in Problem 8.8 for the $R_c = 1/2$, $K = 3$, convolutional code to compute the probability of a bit error for an AWGN channel with

a. Hard-decision decoding
b. Soft-decision decoding

Compare the performance by plotting the results of the computation on the same graph.

8.19 Draw the state diagram for the convolutional code generated by the encoder shown in Figure P8.19, and thus determine whether the code is catastrophic. Also, give an example of a rate 1/2, $K = 4$, convolutional encoder that exhibits catastrophic error propagation.

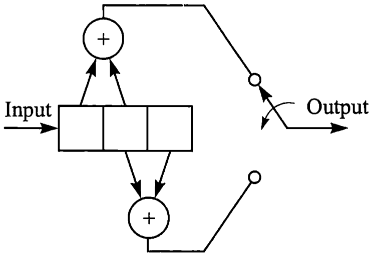


FIGURE P8.19

8.20 A trellis-coded signal is formed as shown in Figure P8.20 by encoding 1 bit by use of a rate 1/2 convolutional code, while 3 additional information bits are left uncoded. Perform the set partitioning of a 32-QAM (cross) constellation, and indicate the subsets in the partition. By how much is the distance between adjacent signal points increased as a result of partitioning?

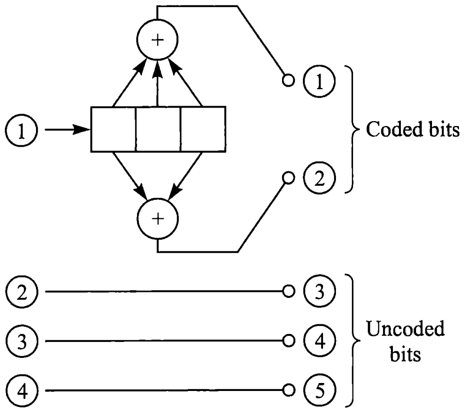


FIGURE P8.20

8.21 Prove Equation 8.4–4.

8.22 Prove that for all real numbers x, y, and z we have

$$\max{}^*\{x, y\} = \max\{x, y\} + \ln(1 + e^{-|x-y|})$$
$$\max{}^*\{x, y, z\} = \max{}^*\{\max{}^*\{x, y\}, z\}$$

8.23 A recursive systematic convolutional code is characterized by

$$\boldsymbol{G}(D) = \begin{bmatrix} 1 & \frac{1}{D+1} \end{bmatrix}$$

This code is used with antipodal signaling with $\mathcal{E}_c = \pm 1$ over an additive white Gaussian noise channel with noise power spectral density of $\frac{N_0}{2} = 2$ W/Hz. It is assumed that the convolutional code is terminated at the zero state and the received sequence is given by

$$\boldsymbol{r} = (0.3, 0.2, 1, -1.2, 1.21.7, 0.3 - 0.6)$$

1. Use the BCJR algorithm to determine the information sequence $\boldsymbol{u}$.
2. Use the Viterbi algorithm to determine the information sequence $\boldsymbol{u}$.

8.24 Apply the Max-Log-APP algorithm to Problem 8.23, and compare the result with the result when the BCJR is used.

8.25 Let X_i, $1 \le i \le n$, denote a sequence of independent binary random variables, and let $p_i(0)$ and $p_i(1)$ denote the probabilities that X_i is equal to 0 and 1, respectively. Let

$$Y = \sum_{i=1}^{n} X_i$$

where the addition is modulo-2, and denote by $p(0)$ and $p(1)$ the probabilities that Y is 0 and 1, respectively.

1. Show that

$$p(0) - p(1) = \prod_{i=1}^{n} (p_i(0) - p_i(1))$$

2. Show that

$$p(0) = \frac{1}{2} + \frac{1}{2}\prod_{i=1}^{n}(p_i(0) - p_i(1))$$

$$p(1) = \frac{1}{2} - \frac{1}{2}\prod_{i=1}^{n}(p_i(0) - p_i(1))$$

3. Using these results, prove Equation 8.10–27.

8.26 Prove Equation 8.10–31 for the equality constraint nodes.

8.27 The parity check matrix of a (12, 3) LDPC code is given by

$$\boldsymbol{H} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Sketch the Tanner graph for this code.

8.28 Show that any $(n, 1)$ repetition code is a LDPC code. Determine the general form of the parity check matrix for an $(n, 1)$ repetition code.

8.29 Sketch the Tanner graph of a (6, 1) repetition code.

9

Digital Communication Through Band-Limited Channels

In previous chapters, we considered the transmission of digital information through an additive Gaussian noise channel. In effect, no bandwidth constraint was imposed on the signal design and the communication system design.

In this chapter, we consider the problem of signal design when the channel is band-limited to some specified bandwidth of W Hz. Under this condition, the channel may be modeled as a linear filter having an equivalent lowpass[†] frequency response $C(f)$ that is zero for $|f| > W$.

The first topic that is treated is the design of the signal pulse $g(t)$ in a linearly modulated signal, represented as

$$v(t) = \sum_n I_n g(t - nT)$$

that efficiently utilizes the total available channel bandwidth W. We shall see that when the channel is ideal for $|f| \leq W$, a signal pulse can be designed that allows us to transmit at symbol rates comparable to or exceeding the channel bandwidth W. On the other hand, when the channel is not ideal, signal transmission at a symbol rate equal to or exceeding W results in intersymbol interference (ISI) among a number of adjacent symbols.

The second topic that we consider is the design of the receiver in the presence of intersymbol interference and AWGN. The solution to the ISI problem is to design a receiver that employs a means for compensating or reducing the ISI in the received signal. The compensator for the ISI is called an equalizer.

We begin our discussion with a general characterization of band-limited linear filter channels.

[†] For convenience, the subscript on lowpass equivalent signals is omitted throughout this chapter.

9.1
CHARACTERIZATION OF BAND-LIMITED CHANNELS

Of the various channels available for digital communications, telephone channels are by far the most widely used. Such channels are characterized as *band-limited linear filters*. This is certainly the proper characterization when frequency-division multiplexing (FDM) is used as a means for establishing channels in the telephone network. Modern telephone networks employ pulse-code modulation (PCM) for digitizing and encoding the analog signal and time-division multiplexing (TDM) for establishing multiple channels. Nevertheless, filtering is still used on the analog signal prior to sampling and encoding. Consequently, even though the present telephone network employs a mixture of FDM and TDM for transmission, the linear filter model for telephone channels is still appropriate.

For our purposes, a bandlimited channel such as a telephone channel will be characterized as a linear filter having an equivalent lowpass frequency-response characteristic $C(f)$. Its equivalent lowpass impulse response is denoted by $c(t)$. Then, if a signal of the form

$$s(t) = \mathrm{Re}\,[v(t)e^{j2\pi f_c t}] \tag{9.1–1}$$

is transmitted over a bandpass telephone channel, the equivalent low-pass received signal is

$$r(t) = \int_{-\infty}^{\infty} v(\tau)c(t-\tau)\,d\tau + z(t) \tag{9.1–2}$$

where the integral represents the convolution of $c(t)$ with $v(t)$, and $z(t)$ denotes the additive noise. Alternatively, the signal term can be represented in the frequency domain as $V(f)C(f)$, where $V(f)$ is the Fourier transform of $v(t)$.

If the channel is band-limited to W Hz, then $C(f) = 0$ for $|f| > W$. As a consequence, any frequency components in $V(f)$ above $|f| = W$ will not be passed by the channel. For this reason, we limit the bandwidth of the transmitted signal to W Hz also.

Within the bandwidth of the channel, we may express the frequency response $C(f)$ as

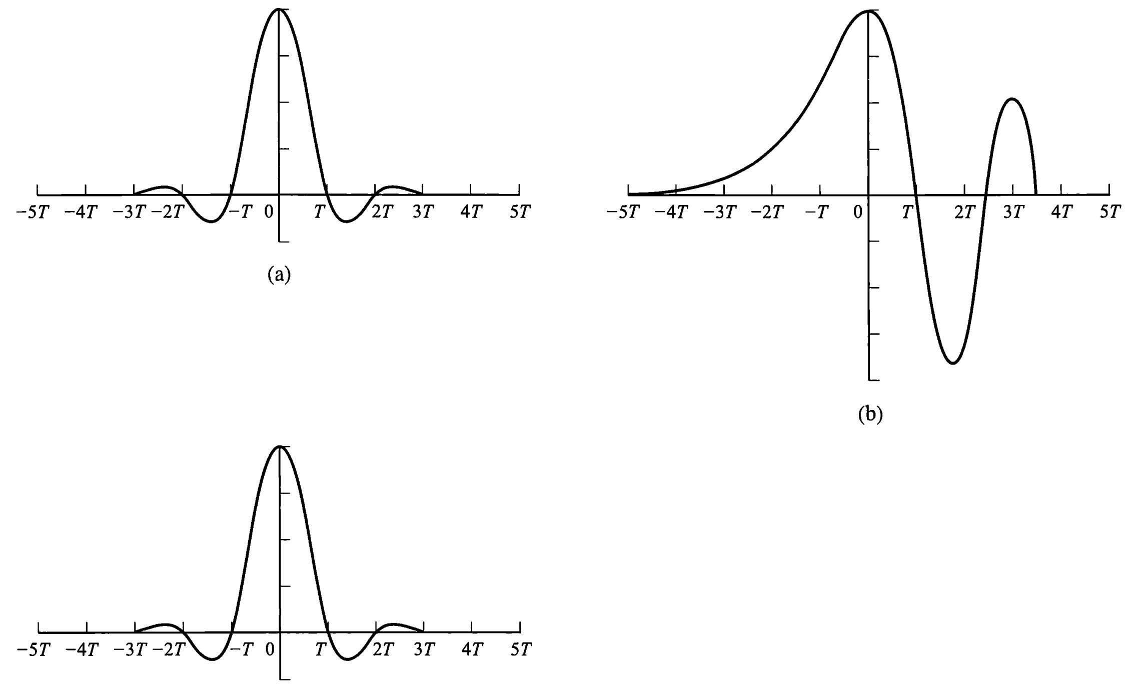
$$C(f) = |C(f)|e^{j\theta(f)} \tag{9.1–3}$$

where $|C(f)|$ is the amplitude-response characteristic and $\theta(f)$ is the phase-response characteristic. Furthermore, the envelope delay characteristic is defined as

$$\tau(f) = -\frac{1}{2\pi}\frac{d\theta(f)}{df} \tag{9.1–4}$$

A channel is said to be *nondistorting* or *ideal* if the amplitude response $|C(f)|$ is constant for all $|f| \le W$ and $\theta(f)$ is a linear function of frequency, i.e., $\tau(f)$ is a constant for all $|f| \le W$. On the other hand, if $|C(f)|$ is not constant for all $|f| \le W$, we say that the channel *distorts the transmitted signal* $V(f)$ *in amplitude*, and, if $\tau(f)$ is not constant for all $|f| \le W$, we say that the channel *distorts the signal* $V(f)$ *in delay*.

As a result of the amplitude and delay distortion caused by the nonideal channel frequency-response characteristic $C(f)$, a succession of pulses transmitted through the channel at rates comparable to the bandwidth W are smeared to the point that they are



(a)

(b)

(c)

FIGURE 9.1–1
Effect of channel distortion: (a) channel input; (b) channel output; (c) equalizer output.

no longer distinguishable as well-defined pulses at the receiving terminal. Instead, they overlap, and, thus, we have intersymbol interference. As an example of the effect of delay distortion on a transmitted pulse, Figure 9.1–1a illustrates a band-limited pulse having zeros periodically spaced in time at points labeled $\pm T$, $\pm 2T$, etc. If information is conveyed by the pulse amplitude, as in PAM, for example, then one can transmit a sequence of pulses, each of which has a peak at the periodic zeros of the other pulses. However, transmission of the pulse through a channel modeled as having a linear envelope delay characteristic $\tau(f)$ (quadratic phase $\theta(f)$) results in the received pulse shown in Figure 9.1–1b having zero-crossings that are no longer periodically spaced. Consequently, a sequence of successive pulses would be smeared into one another and the peaks of the pulses would no longer be distinguishable. Thus, the channel delay distortion results in intesymbol interference. As will be discussed in this chapter, it is possible to compensate for the nonideal frequency-response characteristic of the channel by use of a filter or equalizer at the demodulator. Figure 9.1–1c illustrates the output of a linear equalizer that compensates for the linear distortion in the channel.

The extent of the intersymbol interference on a telephone channel can be appreciated by observing a frequency-response characteristic of the channel. Figure 9.1–2 illustrates the measured average amplitude and delay as functions of frequency for a medium-range (180–725 mi) telephone channel of the switched telecommunications network as given by Duffy and Tratcher (1971). We observe that the usable band of the channel extends from about 300 Hz to about 3000 Hz. The corresponding impulse response of this average channel is shown in Figure 9.1–3. Its duration is about 10 ms. In comparison, the transmitted symbol rates on such a channel may be of the order

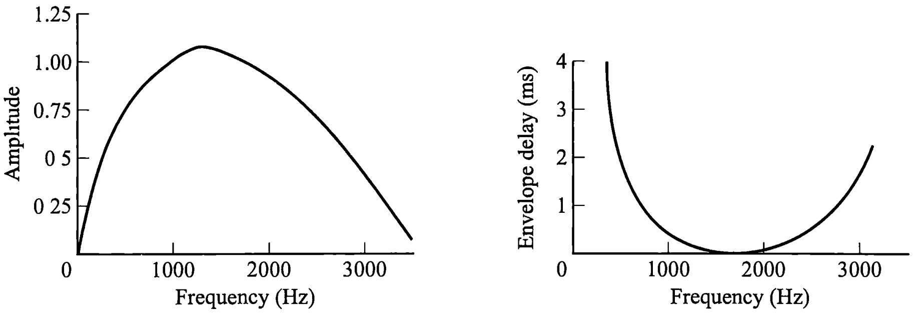


FIGURE 9.1–2
Average amplitude and delay characteristics of medium-range telephone channel.

of 2500 pulses or symbols per second. Hence, intersymbol interference might extend over 20–30 symbols.

In addition to linear distortion, signals transmitted through telephone channels are subject to other impairments, specifically non-linear distortion, frequency offset, phase jitter, impulse noise, and thermal noise.

Non-linear distortion in telephone channels arises from non-linearities in amplifiers and compandors used in the telephone system. This type of distortion is usually small and it is very difficult to correct.

A small *frequency offset*, usually less than 5 Hz, results from the use of carrier equipment in the telephone channel. Such an offset cannot be tolerated in high-speed digital transmission systems that use synchronous phase-coherent demodulation. The offset is usually compensated for by the carrier recovery loop in the demodulator.

Phase jitter is basically a low-index frequency modulation of the transmitted signal with the low-frequency harmonics of the power line frequency (50–60 Hz). Phase jitter poses a serious problem in digital transmission at high rates. However, it can be tracked and compensated for, to some extent, at the demodulator.

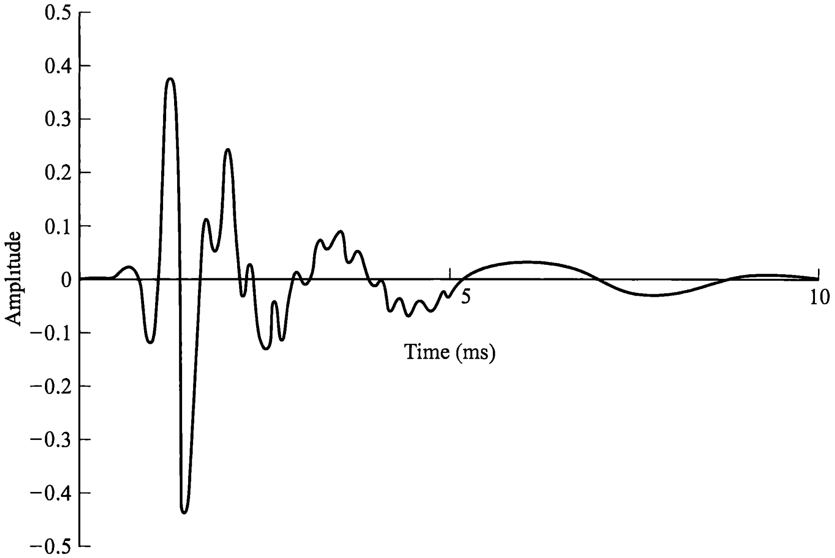


FIGURE 9.1–3
Impulse response of average channel with amplitude and delay shown in Figure 9.1–2.

Impulse noise is an additive disturbance. It arises primarily from the switching equipment in the telephone system. *Thermal* (Gaussian) *noise* is also present at levels of 30 dB or more below the signal.

The degree to which one must be concerned with these channel impairments depends on the transmission rate over the channel and the modulation technique. For rates below 1800 bits/s ($R/W < 1$), one can choose a modulation technique, e.g., FSK, that is relatively insensitive to the amount of distortion encountered on typical telephone channels from all the sources listed above. For rates between 1800 and 2400 bits/s ($R/W \approx 1$), a more bandwidth-efficient modulation technique such as four-phase PSK is usually employed. At these rates, some form of compromise equalization is often employed to compensate for the average amplitude and delay distortion in the channel. In addition, the carrier recovery method is designed to compensate for the frequency offset. The other channel impairments are not that serious in their effects on the error rate performance at these rates. At transmission rates above 2400 bits/s ($R/W > 1$), bandwidth-efficient coded modulation techniques such as trellis-coded QAM, PAM, and PSK are employed. For such rates, special attention must be paid to linear distortion, frequency offset, and phase jitter. Linear distortion is usually compensated for by means of an adaptive equalizer. Phase jitter is handled by a combination of signal design and some type of phase compensation at the demodulator. At rates above 9600 bits/s, special attention must be paid not only to linear distortion, phase jitter, and frequency offset, but also to the other channel impairments mentioned above.

Unfortunately, a channel model that encompasses all the impairments listed above becomes difficult to analyze. For mathematical tractability the channel model that is adopted in this and the next chapter is a linear filter that introduces amplitude and delay distortion and adds Gaussian noise.

Besides the telephone channels, there are other physical channels that exhibit some form of time dispersion and, thus, introduce intersymbol interference. Radio channels such as shortwave ionospheric channels (HF), tropospheric scatter channels, and mobile radio channels are examples of time-dispersive channels. In these channels, time dispersion and, hence, intersymbol interference are the result of multiple propagation paths with different path delays. The number of paths and the relative time delays among the paths vary with time, and, for this reason, these radio channels are usually called *time-variant multipath channels*. The time-variant multipath conditions give rise to a wide variety of frequency-response characteristics. Consequently the frequency-response characterization that is used for telephone channels is inappropriate for time-variant multipath channels. Instead, these radio channels are characterized statistically, as explained in more detail in Chapter 13, in terms of the scattering function, which, in brief, is a two-dimensional representation of the average received signal power as a function of relative time delay and Doppler frequency.

In this chapter, we deal exclusively with the linear time-invariant filter model for a band-limited channel. The adaptive equalization techniques presented in Chapter 10 for combating intersymbol interference are also applicable to time-variant multipath channels, under the condition that the time variations in the channel are relatively slow in comparison to the total channel bandwidth or, equivalently, to the symbol transmission rate over the channel.

9.2
SIGNAL DESIGN FOR BAND-LIMITED CHANNELS

It was shown in Chapter 3 that the equivalent lowpass transmitted signal for several different types of digital modulation techniques has the common form

$$v(t) = \sum_{n=0}^{\infty} I_n g(t - nT) \tag{9.2–1}$$

where $\{I_n\}$ represents the discrete information-bearing sequence of symbols and $g(t)$ is a pulse that, for the purposes of this discussion, is assumed to have a band-limited frequency-response characteristic $G(f)$, i.e., $G(f) = 0$ for $|f| > W$. This signal is transmitted over a channel having a frequency response $C(f)$, also limited to $|f| \leq W$. Consequently, the received signal can be represented as

$$r_l(t) = \sum_{n=0}^{\infty} I_n h(t - nT) + z(t) \tag{9.2–2}$$

where

$$h(t) = \int_{-\infty}^{\infty} g(\tau) c(t - \tau)\, d\tau \tag{9.2–3}$$

and $z(t)$ represents the additive white Gaussian noise.

Let us suppose that the received signal is passed first through a filter and then sampled at a rate $1/T$ samples/s. We shall show in a subsequent section that the optimum filter from the point of view of signal detection is one matched to the received pulse. That is, the frequency response of the receiving filter is $H^*(f)$. We denote the output of the receiving filter as

$$y(t) = \sum_{n=0}^{\infty} I_n x(t - nT) + \nu(t) \tag{9.2–4}$$

where $x(t)$ is the pulse representing the response of the receiving filter to the input pulse $h(t)$ and $\nu(t)$ is the response of the receiving filter to the noise $z(t)$.

Now, if $y(t)$ is sampled at times $t = kT + \tau_0$, $k = 0, 1, \ldots,$ we have

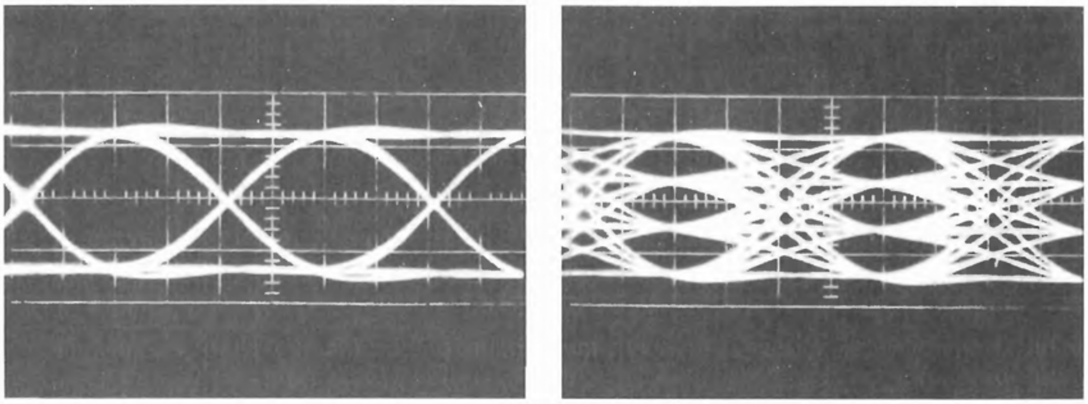
$$y(kT + \tau_0) \equiv y_k = \sum_{n=0}^{\infty} I_n x(kT - nT + \tau_0) + \nu(kT + \tau_0) \tag{9.2–5}$$

or, equivalently,

$$y_k = \sum_{n=0}^{\infty} I_n x_{k-n} + \nu_k, \qquad k = 0, 1, \ldots \tag{9.2–6}$$

where τ_0 is the transmission delay through the channel. The sample values can be expressed as

$$y_k = x_0 \left(I_k + \frac{1}{x_0} \sum_{\substack{n=0 \\ n \neq k}}^{\infty} I_n x_{k-n} \right) + \nu_k, \qquad k = 0, 1, \ldots \tag{9.2–7}$$



(a) Binary (b) Quaternary

FIGURE 9.2–1
Examples of eye patterns for binary and quaternary amplitude-shift keying (or PAM).

We regard x_0 as an arbitrary scale factor, which we arbitrarily set equal to unity for convenience. Then

$$y_k = I_k + \sum_{\substack{n=0 \\ n\neq k}}^{\infty} I_n x_{k-n} + \nu_k \tag{9.2–8}$$

The term I_k represents the desired information symbol at the kth sampling instant, the term

$$\sum_{\substack{n=0 \\ n\neq k}}^{\infty} I_n x_{k-n}$$

represents the ISI, and ν_k is the additive Gaussian noise variable at the kth sampling instant.

The amount of intersymbol interference and noise in a digital communication system can be viewed on an oscilloscope. For PAM signals, we can display the received signal $y(t)$ on the vertical input with the horizontal sweep rate set at $1/T$. The resulting oscilloscope display is called an *eye pattern* because of its resemblance to the human eye. For example, Figure 9.2–1 illustrates the eye patterns for binary and four-level PAM modulation. The effect of ISI is to cause the eye to close, thereby reducing the margin for additive noise to cause errors. Figure 9.2–2 graphically illustrates the effect of

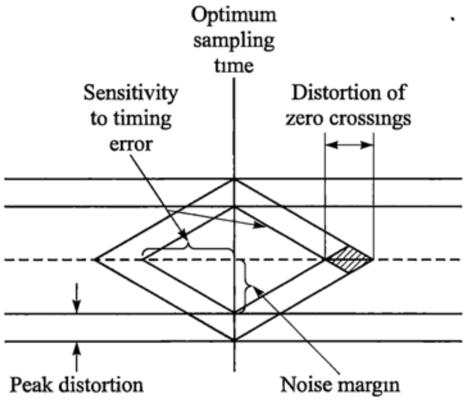


FIGURE 9.2–2
Effect of intersymbol interference on eye opening.

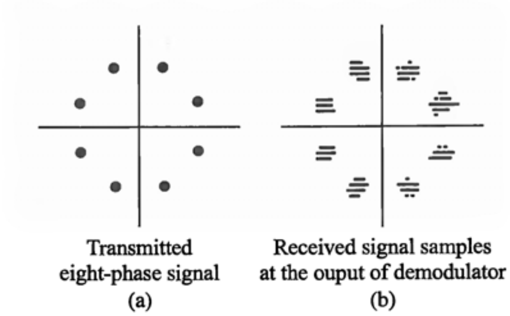


FIGURE 9.2–3
Two-dimensional digital "eye patterns."

intersymbol interference in reducing the opening of a binary eye. Note that intersymbol interference distorts the position of the zero-crossings and causes a reduction in the eye opening. Thus, it causes the system to be more sensitive to a synchronization error.

For PSK and QAM it is customary to display the "eye pattern" as a two-dimensional scatter diagram illustrating the sampled values $\{y_k\}$ that represent the decision variables at the sampling instants. Figure 9.2–3 illustrates such an eye pattern for an 8-PSK signal. In the absence of intersymbol interference and noise, the superimposed signals at the sampling instants would result in eight distinct points corresponding to the eight transmitted signal phases. Intersymbol interference and noise result in a deviation of the received samples $\{y_k\}$ from the desired 8-PSK signal. The larger the intersymbol interference and noise, the larger the scattering of the received signal samples relative to the transmitted signal points.

Below, we consider the problem of signal design under the condition that there is no intersymbol interference at the sampling instants.

9.2–1 Design of Band-Limited Signals for No Intersymbol Interference—The Nyquist Criterion

For the discussion in this section and in Section 9.2–2, we assume that the band-limited channel has ideal frequency-response characteristics, i.e., $C(f) = 1$ for $|f| \leq W$. Then the pulse $x(t)$ has a spectral characteristic $X(f) = |G(f)|^2$, where

$$x(t) = \int_{-W}^{W} X(f) e^{j2\pi f t}\, df \tag{9.2–9}$$

We are interested in determining the spectral properties of the pulse $x(t)$ and, hence, the transmitted pulse $g(t)$, that results in no intersymbol interference. Since

$$y_k = I_k + \sum_{\substack{n=0 \\ n\neq k}}^{\infty} I_n x_{k-n} + \nu_k \tag{9.2–10}$$

the condition for no intersymbol interference is

$$x(t = kT) \equiv x_k = \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{9.2–11}$$

Below, we derive the necessary and sufficient condition on $X(f)$ in order for $x(t)$ to satisfy the above relation. This condition is known as the *Nyquist pulse-shaping criterion* or *Nyquist condition for zero ISI* and is stated in the following theorem.

THEOREM: (NYQUIST). The necessary and sufficient condition for $x(t)$ to satisfy

$$x(nT) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{9.2–12}$$

is that its Fourier transform $X(f)$ satisfy

$$\sum_{m=-\infty}^{\infty} X(f + m/T) = T \tag{9.2–13}$$

Proof. In general, $x(t)$ is the inverse Fourier transform of $X(f)$. Hence,

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft}\, df \tag{9.2–14}$$

At the sampling instants $t = nT$, this relation becomes

$$x(nT) = \int_{-\infty}^{\infty} X(f) e^{j2\pi fnT}\, df \tag{9.2–15}$$

Let us break up the integral in Equation 9.2–15 into integrals covering the finite range of $1/T$. Thus, we obtain

$$\begin{aligned} x(nT) &= \sum_{m=-\infty}^{\infty} \int_{(2m-1)/2T}^{(2m+1)/2T} X(f) e^{j2\pi fnT} df \\ &= \sum_{m=-\infty}^{\infty} \int_{-1/2T}^{1/2T} X(f + m/T) e^{j2\pi fnT} df \\ &= \int_{-1/2T}^{1/2T} \left[\sum_{m=-\infty}^{\infty} X(f + m/T) \right] e^{j2\pi fnT} df \\ &= \int_{-1/2T}^{1/2T} B(f) e^{j2\pi fnT} df \end{aligned} \tag{9.2–16}$$

where we have defined $B(f)$ as

$$B(f) = \sum_{m=-\infty}^{\infty} X(f + m/T) \tag{9.2–17}$$

Obviously $B(f)$ is a periodic function with period $1/T$, and, therefore, it can be expanded in terms of its Fourier series coefficients $\{b_n\}$ as

$$B(f) = \sum_{n=-\infty}^{\infty} b_n e^{j2\pi nfT} \tag{9.2–18}$$

where

$$b_n = T \int_{-1/2T}^{1/2T} B(f) e^{-j2\pi nfT} df \tag{9.2–19}$$

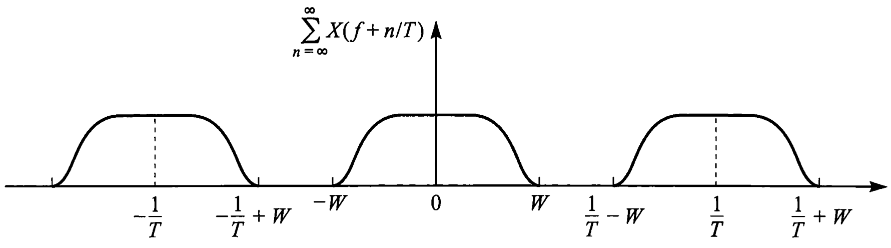


FIGURE 9.2–4
Plot of $B(f)$ for the case $T < 1/2W$.

Comparing Equations 9.2–19 and 9.2–16, we obtain

$$b_n = Tx(-nT) \tag{9.2–20}$$

Therefore, the necessary and sufficient condition for Equation 9.2–11 to be satisfied is that

$$b_n = \begin{cases} T & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{9.2–21}$$

which, when substituted into Equation 9.2–18, yields

$$B(f) = T \tag{9.2–22}$$

or, equivalently,

$$\sum_{m=-\infty}^{\infty} X(f + m/T) = T \tag{9.2–23}$$

This concludes the proof of the theorem.

Now suppose that the channel has a bandwidth of W. Then $C(f) \equiv 0$ for $|f| > W$ and, consequently, $X(f) = 0$ for $|f| > W$. We distinguish three cases.

1. When $T < 1/2W$, or, equivalently, $1/T > 2W$, since $B(f) = \sum_{n=-\infty}^{+\infty} X(f+n/T)$ consists of nonoverlapping replicas of $X(f)$, separated by $1/T$ as shown in Figure 9.2–4, there is no choice for $X(f)$ to ensure $B(f) \equiv T$ in this case and there is no way that we can design a system with no ISI.
2. When $T = 1/2W$, or, equivalently, $1/T = 2W$ (the Nyquist rate), the replications of $X(f)$, separated by $1/T$, are as shown in Figure 9.2–5. It is clear that in this case

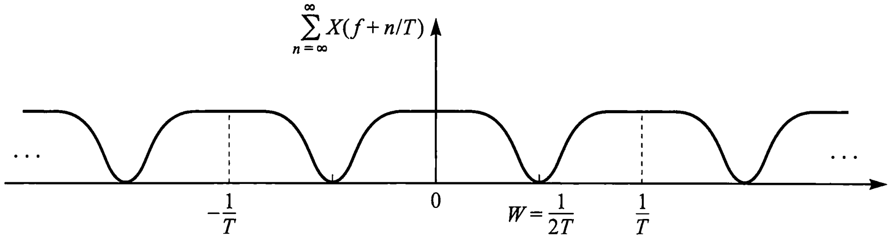


FIGURE 9.2–5
Plot of $B(f)$ for the case $T = 1/2W$.

there exists only one $X(f)$ that results in $B(f) = T$, namely,

$$X(f) = \begin{cases} T & |f| < W \\ 0 & \text{otherwise} \end{cases} \tag{9.2–24}$$

which corresponds to the pulse

$$x(t) = \frac{\sin(\pi t/T)}{\pi t/T} \equiv \operatorname{sinc}\left(\frac{\pi t}{T}\right) \tag{9.2–25}$$

This means that the smallest value of T for which transmission with zero ISI is possible is $T = 1/2W$, and for this value, $x(t)$ has to be a sinc function. The difficulty with this choice of $x(t)$ is that it is noncausal and, therefore, nonrealizable. To make it realizable, usually a delayed version of it, i.e., $\operatorname{sinc}[\pi(t - t_0)/T]$ is used and t_0 is chosen such that for $t < 0$, we have $\operatorname{sinc}[\pi(t - t_0)/T] \approx 0$. Of course, with this choice of $x(t)$, the sampling time must also be shifted to $mT + t_0$. A second difficulty with this pulse shape is that its rate of convergence to zero is slow. The tails of $x(t)$ decay as $1/t$; consequently, a small mistiming error in sampling the output of the matched filter at the demodulator results in an infinite series of ISI components. Such a series is not absolutely summable because of the $1/t$ rate of decay of the pulse, and, hence, the sum of the resulting ISI does not converge.

3. When $T > 1/2W$, $B(f)$ consists of overlapping replications of $X(f)$ separated by $1/T$, as shown in Figure 9.2–6. In this case, there exist numerous choices for $X(f)$ such that $B(f) \equiv T$.

A particular pulse spectrum, for the $T > 1/2W$ case, that has desirable spectral properties and has been widely used in practice is the raised cosine spectrum. The raised cosine frequency characteristic is given as (see Problem 9.16)

$$X_{rc}(f) = \begin{cases} T & 0 \le |f| \le \dfrac{1-\beta}{2T} \\ \dfrac{T}{2}\left\{1 + \cos\left[\dfrac{\pi T}{\beta}\left(|f| - \dfrac{1-\beta}{2T}\right)\right]\right\} & \dfrac{1-\beta}{2T} \le |f| \le \dfrac{1+\beta}{2T} \\ 0 & |f| > \dfrac{1+\beta}{2T} \end{cases} \tag{9.2–26}$$

where β is called the *roll-off factor* and takes values in the range $0 \le \beta \le 1$. The bandwidth occupied by the signal beyond the Nyquist frequency $1/2T$ is called the

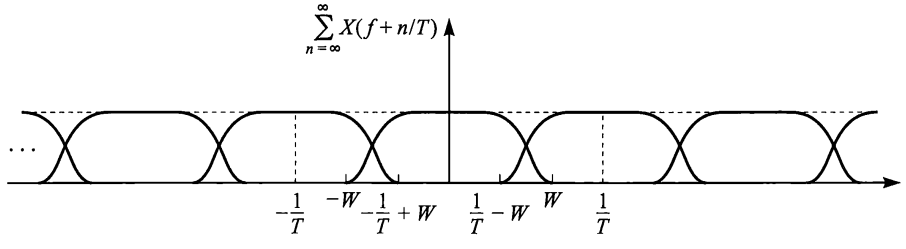


FIGURE 9.2–6
Plot of $B(f)$ for the case $T > 1/2W$.

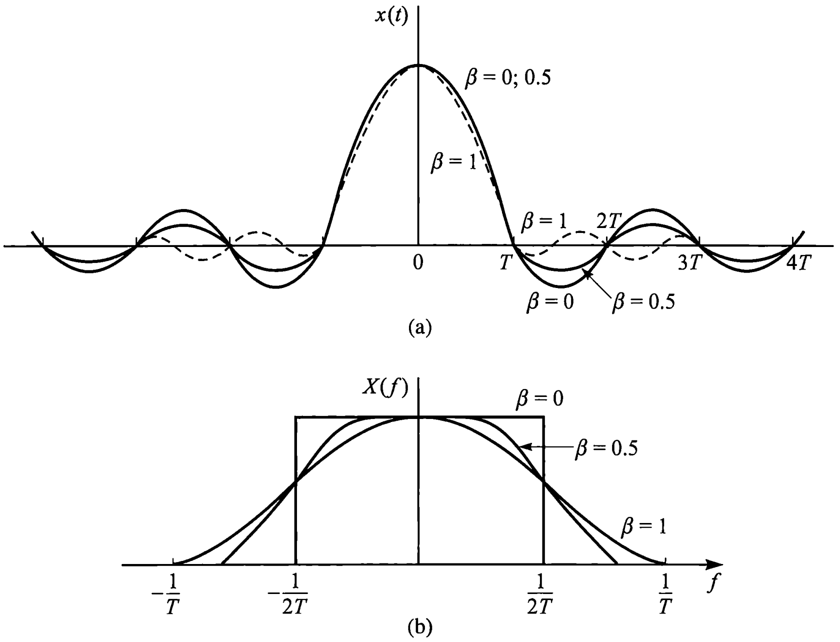


FIGURE 9.2–7
Pulses having a raised cosine spectrum.

excess bandwidth and is usually expressed as a percentage of the Nyquist frequency. For example, when $\beta = \frac{1}{2}$, the excess bandwidth is 50 percent and when $\beta = 1$, the excess bandwidth is 100 percent. The pulse $x(t)$, having the raised cosine spectrum, is

$$\begin{aligned} x(t) &= \frac{\sin(\pi t/T)}{\pi t/T} \frac{\cos(\pi \beta t/T)}{1 - 4\beta^2 t^2/T^2} \\ &= \operatorname{sinc}(\pi t/T) \frac{\cos(\pi \beta t/T)}{1 - 4\beta^2 t^2/T^2} \end{aligned} \tag{9.2–27}$$

Note that $x(t)$ is normalized so that $x(0) = 1$. Figure 9.2–7 illustrates the raised cosine spectral characteristics and the corresponding pulses for $\beta = 0, \frac{1}{2}$, and 1. Note that for $\beta = 0$, the pulse reduces to $x(t) = \operatorname{sinc}(\pi t/T)$, and the symbol rate $1/T = 2W$. When $\beta = 1$, the symbol rate is $1/T = W$. In general, the tails of $x(t)$ decay as $1/t^3$ for $\beta > 0$. Consequently, a mistiming error in sampling leads to a series of ISI components that converges to a finite value.

Because of the smooth characteristics of the raised cosine spectrum, it is possible to design practical filters for the transmitter and the receiver that approximate the overall desired frequency response. In the special case where the channel is ideal, i.e., $C(f) = 1, |f| \leq W$, we have

$$X_{rc}(f) = G_T(f) G_R(f) \tag{9.2–28}$$

where $G_T(f)$ and $G_R(f)$ are the frequency responses of the two filters. In this case, if the receiver filter is matched to the transmitter filter, we have $X_{rc}(f) = G_T(f)G_R(f) = |G_T(f)|^2$. Ideally,

$$G_T(f) = \sqrt{|X_{rc}(f)|}\, e^{-j2\pi f t_0} \tag{9.2–29}$$

and $G_R(f) = G_T^*(f)$, where t_0 is some nominal delay that is required to ensure physical realizability of the filter. Thus, the overall raised cosine spectral characteristic is split evenly between the transmitting filter and the receiving filter. Note also that an additional delay is necessary to ensure the physical realizability of the receiving filter.

9.2–2 Design of Band-Limited Signals with Controlled ISI—Partial-Response Signals

As we have observed from our discussion of signal design for zero ISI, it is necessary to reduce the symbol rate $1/T$ below the Nyquist rate of $2W$ symbols/s to realize practical transmitting and receiving filters. On the other hand, suppose we choose to relax the condition of zero ISI and, thus, achieve a symbol transmission rate of $2W$ symbols/s. By allowing for a controlled amount of ISI, we can achieve this symbol rate.

We have already seen that the condition for zero ISI is $x(nT) = 0$ for $n \neq 0$. However, suppose that we design the band-limited signal to have controlled ISI at one time instant. This means that we allow one additional nonzero value in the samples $\{x(nT)\}$. The ISI that we introduce is deterministic or "controlled" and, hence, it can be taken into account at the receiver, as discussed below.

One special case that leads to (approximately) physically realizable transmitting and receiving filters is specified by the samples[†]

$$x(nT) = \begin{cases} 1 & n = 0, 1 \\ 0 & \text{otherwise} \end{cases} \tag{9.2–30}$$

Now, using Equation 9.2–20, we obtain

$$b_n = \begin{cases} T & n = 0, -1 \\ 0 & \text{otherwise} \end{cases} \tag{9.2–31}$$

which, when substituted into Equation 9.2–18, yields

$$B(f) = T + Te^{-j2\pi fT} \tag{9.2–32}$$

As in the preceding section, it is impossible to satisfy the above equation for $T < 1/2W$. However, for $T = 1/2W$, we obtain

$$\begin{aligned} X(f) &= \begin{cases} \dfrac{1}{2W}(1 + e^{-j\pi f/W}) & |f| < W \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} \dfrac{1}{W} e^{-j\pi f/2W} \cos \dfrac{\pi f}{2W} & |f| < W \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{9.2–33}$$

[†] It is convenient to deal with samples of $x(t)$ that are normalized to unity for $n = 0, 1$.

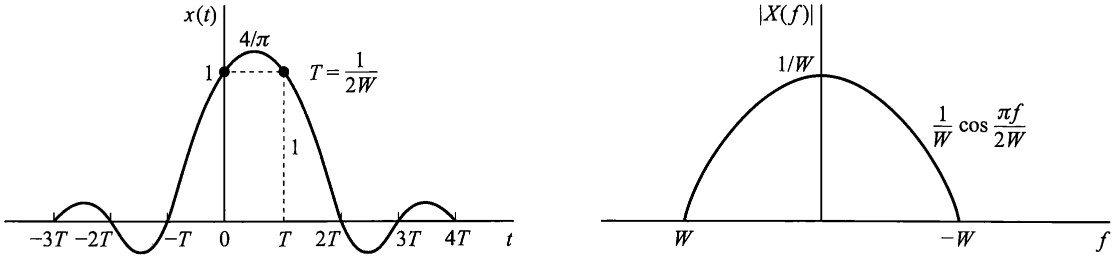


FIGURE 9.2–8
Time-domain and frequency-domain characteristics of a duobinary signal.

Therefore, $x(t)$ is given by

$$x(t) = \operatorname{sinc}(2\pi Wt) + \operatorname{sinc}\left[2\pi\left(Wt - \frac{1}{2}\right)\right] \tag{9.2–34}$$

This pulse is called a *duobinary signal pulse* . It is illustrated along with its magnitude spectrum in Figure 9.2–8. Note that the spectrum decays to zero smoothly, which means that physically realizable filters can be designed that approximate this spectrum very closely. Thus, a symbol rate of $2W$ is achieved.

Another special case that leads to (approximately) physically realizable transmitting and receiving filters is specified by the samples

$$x\left(\frac{n}{2W}\right) = x(nT) = \begin{cases} 1 & n = -1 \\ -1 & n = 1 \\ 0 & \text{otherwise} \end{cases} \tag{9.2–35}$$

The corresponding pulse $x(t)$ is given as

$$x(t) = \operatorname{sinc}\frac{\pi(t+T)}{T} - \operatorname{sinc}\frac{\pi(t-T)}{T} \tag{9.2–36}$$

and its spectrum is

$$X(f) = \begin{cases} \dfrac{1}{2W}\left(e^{j\pi f/W} - e^{-j\pi f/W}\right) = \dfrac{j}{W}\sin\dfrac{\pi f}{W} & |f| \le W \\ 0 & |f| > W \end{cases} \tag{9.2–37}$$

This pulse and its magnitude spectrum are illustrated in Figure 9.2–9. It is called a *modified duobinary signal pulse.* It is interesting to note that the spectrum of this signal has a zero at $f = 0$, making it suitable for transmission over a channel that does not pass DC.

One can obtain other interesting and physically realizable filter characteristics, as shown by Kretzmer (1966) and Lucky et al. (1968), by selecting different values for the samples $\{x(n/2W)\}$ and more than two nonzero samples. However, as we select more nonzero samples, the problem of unraveling the controlled ISI becomes more cumbersome and impractical.

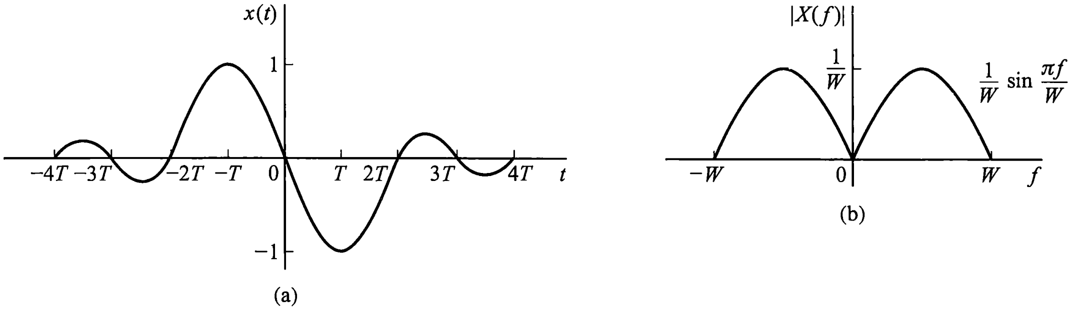


FIGURE 9.2–9
Time-domain and frequency-domain characteristics of a modified duobinary signal.

In general, the class of band-limited signal pulses that have the form

$$x(t) = \sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right) \text{sinc}\left[2\pi W\left(t - \frac{n}{2W}\right)\right] \tag{9.2–38}$$

and their corresponding spectra

$$X(f) = \begin{cases} \dfrac{1}{2W} \displaystyle\sum_{n=-\infty}^{\infty} x\left(\frac{n}{2W}\right) e^{-jn\pi f/W} & |f| \le W \\ 0 & |f| > W \end{cases} \tag{9.2–39}$$

are called *partial-response signals* when controlled ISI is a purposely introduced by selecting two or more nonzero samples from the set $\{x(n/2W)\}$. The resulting signal pulses allow us to transmit information symbols at the Nyquist rate of $2W$ symbols/s. The detection of the received symbols in the presence of controlled ISI is described below.

Alternative characterization of partial-response signals We conclude this subsection by presenting another interpretation of a partial-response signal. Suppose that the partial-response signal is generated, as shown in Figure 9.2–10, by passing the discrete-time sequence $\{I_n\}$ through a discrete-time filter with coefficients $x_n \equiv x(n/2W)$, $n = 0, 1, \ldots, N-1$, and using the output sequence $\{B_n\}$ from this filter to excite periodically with an input $B_n\delta(t - nT)$ an analog filter having an impulse response $\text{sinc}(2\pi Wt)$. The resulting output signal is identical to the partial-response signal given by Equation 9.2–38.

Since

$$B_n = \sum_{k=0}^{N-1} x_k I_{n-k} \tag{9.2–40}$$

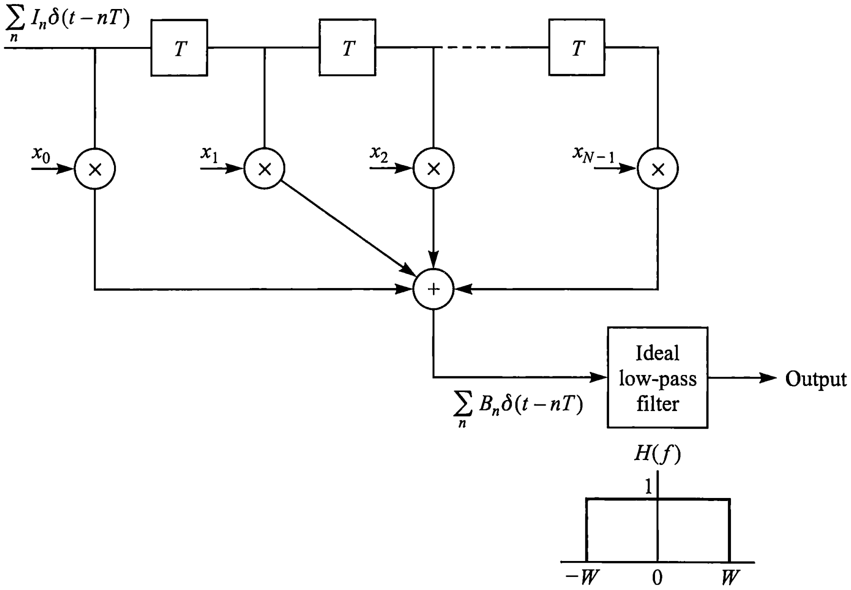


FIGURE 9.2–10
An alternative method for generating a partial-response signal.

the sequence of symbols $\{B_n\}$ is correlated as a consequence of the filtering performed on the sequence $\{I_n\}$. In fact, the autocorrelation function of the sequence $\{B_n\}$ is

$$\begin{aligned} R(m) &= E(B_n B_{n+m}) \\ &= \sum_{k=0}^{N-1}\sum_{l=0}^{N-1} x_k x_l E(I_{n-k} I_{n+m-l}) \end{aligned} \tag{9.2–41}$$

When the input sequence is zero-mean and white,

$$E(I_{n-k} I_{n+m-l}) = \delta_{m+k-l} \tag{9.2–42}$$

where we have used the normalization $E\left(I_n^2\right) = 1$. Substitution of Equation 9.2–42, into Equation 9.2–41 yields the desired autocorrelation function for $\{B_n\}$ in the form

$$R(m) = \sum_{k=0}^{N-1-|m|} x_k x_{k+|m|}, \qquad m = 0, \pm 1, \ldots, \pm(N-1) \tag{9.2–43}$$

The corresponding power spectral density is

$$\begin{aligned} \mathcal{S}(f) &= \sum_{m=-(N-1)}^{N-1} R(m) e^{-j2\pi fmT} \\ &= \left| \sum_{m=0}^{N-1} x_m e^{-j2\pi fmT} \right|^2 \end{aligned} \tag{9.2–44}$$

where $T = 1/2W$ and $|f| \leq 1/2T = W$. Thus, the partial-response signal designs provide spectral shaping of the signal transmitted through the channel.

9.2–3 Data Detection for Controlled ISI

In this section, we describe two methods for detecting the information symbols at the receiver when the received signal contains controlled ISI. One is a symbol-by-symbol detection method that is relatively easy to implement. The second method is based on the maximum-likelihood criterion for detecting a sequence of symbols. The latter method minimizes the probability of error but is a little more complex to implement. In particular, we consider the detection of the duobinary and the modified duobinary partial-response signals. In both cases, we assume that the desired spectral characteristic $X(f)$ for the partial-response signal is split evenly between the transmitting and receiving filters, i.e., $|G_T(f)| = |G_R(f)| = |X(f)|^{1/2}$. This treatment is based on PAM signals, but it is easily generalized to QAM and PSK.

Symbol-by-symbol suboptimum detection For the duobinary signal pulse, $x(nT) = 1$, for $n = 0, 1$, and is zero otherwise. Hence, the samples at the output of the receiving filter (demodulator) have the form

$$y_m = B_m + \nu_m = I_m + I_{m-1} + \nu_m \tag{9.2–45}$$

where $\{I_m\}$ is the transmitted sequence of amplitudes and $\{\nu_m\}$ is a sequence of additive Gaussian noise samples. Let us ignore the noise for the moment and consider the binary case where $I_m = \pm 1$ with equal probability. Then B_m takes on one of three possible values, namely, $B_m = -2, 0, 2$ with corresponding probabilities 1/4, 1/2, 1/4. If I_{m-1} is the detected symbol from the $(m-1)$th signaling interval, its effect on B_m, the received signal in the mth signaling interval, can be eliminated by subtraction, thus allowing I_m to be detected. This process can be repeated sequentially for every received symbol.

The major problem with this procedure is that errors arising from the additive noise tend to propagate. For example, if I_{m-1} is in error, its effect on B_m is not eliminated but, in fact, is reinforced by the incorrect subtraction. Consequently, the detection of I_m is also likely to be in error.

Error propagation can be avoided by *precoding* the data at the transmitter instead of eliminating the controlled ISI by subtraction at the receiver. The precoding is performed on the binary data sequence prior to modulation. From the data sequence $\{D_n\}$ of 1s and 0s that is to be transmitted, a new sequence $\{P_n\}$, called the *precoded sequence*, is generated. For the duobinary signal, the precoded sequence is defined as

$$P_m = D_m \ominus P_{m-1}, \qquad m = 1, 2, \ldots \tag{9.2–46}$$

where $\ominus$ denotes modulo-2 subtraction.† Then we set $I_m = -1$ if $P_m = 0$ and $I_m = 1$ if $P_m = 1$, i.e., $I_m = 2P_m - 1$. Note that this precoding operation is identical to that described in Section 3.3 in the context of our discussion of an NRZI signal.

---

†Although this is identical to modulo-2 addition, it is convenient to view the precoding operation for duobinary in terms of modulo-2 subtraction.

The noise-free samples at the output of the receiving filter are given by

$$\begin{aligned} B_m &= I_m + I_{m-1} \\ &= (2P_m - 1) + (2P_{m-1} - 1) \\ &= 2(P_m + P_{m-1} - 1) \end{aligned} \tag{9.2–47}$$

Consequently,

$$P_m + P_{m-1} = \tfrac{1}{2}B_m + 1 \tag{9.2–48}$$

Since $D_m = P_m \oplus P_{m-1}$, it follows that the data sequence D_m is obtained from B_m using the relation

$$D_m = \tfrac{1}{2}B_m + 1 \quad (\text{mod } 2) \tag{9.2–49}$$

Consequently, if $B_m = \pm 2$, then $D_m = 0$, and if $B_m = 0$, then $D_m = 1$. An example that illustrates the precoding and decoding operations is given in Table 9.2–1. In the presence of additive noise, the sampled outputs from the receiving filter are given by Equation 9.2–45. In this case $y_m = B_m + \nu_m$ is compared with the two thresholds set at $+1$ and -1. The data sequence $\{D_n\}$ is obtained according to the detection rule

$$D_m = \begin{cases} 1 & (|y_m| < 1) \\ 0 & (|y_m| \ge 1) \end{cases} \tag{9.2–50}$$

The extension from binary PAM to multilevel PAM signaling using the duobinary pulses is straightforward. In this case the M-level amplitude sequence $\{I_m\}$ results in a (noise-free) sequence

$$B_m = I_m + I_{m-1}, \qquad m = 1, 2, \ldots \tag{9.2–51}$$

which has $2M - 1$ possible equally spaced levels. The amplitude levels are determined from the relation

$$I_m = 2P_m - (M - 1) \tag{9.2–52}$$

TABLE 9.2–1
Binary Signaling with Duobinary Pulses

Data sequence D_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1
Precoded sequence P_n	0	1	0	1	1	0	0	0	1	1	1	1	0	1	1	0
Transmitted sequence I_n	−1	1	−1	1	1	−1	−1	−1	1	1	1	1	−1	1	1	−1
Received sequence B_n		0	0	0	2	0	−2	−2	0	2	2	2	0	0	2	0
Decoded sequence D_n		1	1	1	0	1	0	0	1	0	0	0	1	1	0	1

where $\{P_m\}$ is the precoded sequence that is obtained from an M-level data sequence $\{D_m\}$ according to the relation

$$P_m = D_m \ominus P_{m-1} \quad (\text{mod } M) \tag{9.2–53}$$

where the possible values of the sequence $\{D_m\}$ are $0, 1, 2, \ldots, M-1$.

In the absence of noise, the samples at the output of the receiving filter may be expressed as

$$B_m = I_m + I_{m-1} = 2[P_m + P_{m-1} - (M-1)] \tag{9.2–54}$$

Hence,

$$P_m + P_{m-1} = \tfrac{1}{2}B_m + (M-1) \tag{9.2–55}$$

Since $D_m = P_m + P_{m-1}$ (mod M), it follows that

$$D_m = \tfrac{1}{2}B_m + (M-1) \quad (\text{mod } M) \tag{9.2–56}$$

An example illustrating multilevel precoding and decoding is given in Table 9.2–2.

In the presence of noise, the received signal-plus-noise is quantized to the nearest of the possible signal levels and the rule given above is used on the quantized values to recover the data sequence.

In the case of the modified duobinary pulse, the controlled ISI is specified by the values $x(n/2W) = -1$, for $n = 1$, $x(n/2W) = 1$ for $n = -1$, and zero otherwise. Consequently, the noise-free sampled output from the receiving filter is given as

$$B_m = I_m - I_{m-2} \tag{9.2–57}$$

where the M-level sequence $\{I_m\}$ is obtained by mapping a precoded sequence according to the Equation 9.2–52 and

$$P_m = D_m \oplus P_{m-2} \quad (\text{mod } M) \tag{9.2–58}$$

TABLE 9.2–2
Four-Level Signal Transmission with Duobinary Pulses

Data sequence D_m		0	0	1	3	1	2	0	3	3	2	0	1	0
Precoded sequence P_m	0	0	0	1	2	3	3	1	2	1	1	3	2	2
Transmitted sequence I_m	−3	−3	−3	−1	1	3	3	−1	1	−1	−1	3	1	1
Received sequence B_n		−6	−6	−4	0	4	6	2	0	0	−2	2	4	2
Decoded sequence D_m		0	0	1	3	1	2	0	3	3	2	0	1	0

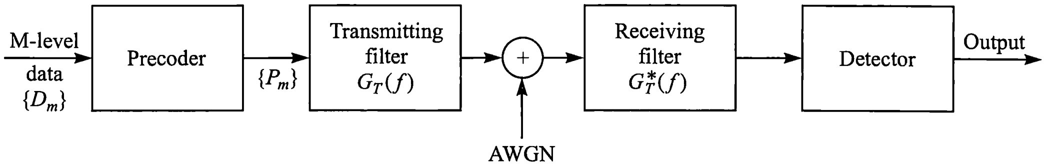


FIGURE 9.2–11
Block diagram of modulator and demodulator for partial-response signals.

From these relations, it is easy to show that the detection rule for recovering the data sequence $\{D_m\}$ from $\{B_m\}$ in the absence of noise is

$$D_m = \tfrac{1}{2}B_m \quad (\text{mod } M) \tag{9.2–59}$$

As demonstrated above, the precoding of the data at the transmitter makes it possible to detect the received data on a symbol-by-symbol basis without having to look back at previously detected symbols. Thus, error propagation is avoided.

The symbol-by-symbol detection rule described above is not the optimum detection scheme for partial-response signals due to the memory inherent in the received signal. Nevertheless, symbol-by-symbol detection is relatively simple to implement and is used in many practical applications involving duobinary and modified duobinary pulse signals.

Let us determine the probability of error for detection of digital M-ary PAM signaling using duobinary and modified duobinary pulses. The channel is assumed to be an ideal band-limited channel with additive white Gaussian noise. The model for the communication system is shown in Figure 9.2–11.

At the transmitter, the M-level data sequence $\{D_m\}$ is precoded as described previously. The precoder output is mapped into one of M possible amplitude levels. Then the transmitting filter with frequency response $G_T(f)$ has an output

$$v(t) = \sum_{n=-\infty}^{\infty} I_n g_T(t - nT) \tag{9.2–60}$$

The partial-respone function $X(f)$ is divided equally between the transmitting and receiving filters. Hence, the receiving filter is matched to the transmitted pulse, and the cascade of the two filters results in the frequency characteristic

$$|G_T(f)G_R(f)| = |X(f)| \tag{9.2–61}$$

The matched filter output is sampled at $t = nT = n/2W$ and the samples are fed to the decoder. For the duobinary signal, the output of the matched filter at the sampling instant may be expressed as

$$y_m = I_m + I_{m-1} + \nu_m = B_m + \nu_m \tag{9.2–62}$$

where ν_m is the additive noise component. Similarly, the output of the matched filter for the modified duobinary signal is

$$y_m = I_m - I_{m-2} + \nu_m = B_m + \nu_m \tag{9.2–63}$$

For binary transmission, let $I_m = \pm d$, where $2d$ is the distance between signal levels. Then, the corresponding values of B_m are $(2d, 0, -2d)$. For M-ary PAM signal transmission, where $I_m = \pm d, \pm 3d, \ldots, \pm(M-1)d$, the received signal levels are $B_m = 0, \pm 2d, \pm 4d, \ldots, \pm 2(M-1)d$. Hence, the number of received levels is $2M - 1$, and the scale factor d is equivalent to $x_0 = \mathcal{E}_g$.

The input transmitted symbols $\{I_m\}$ are assumed to be equally probable. Then, for duobinary and modified duobinary signals, it is easily demonstrated that, in the absence of noise, the received output levels have a (triangular) probability distribution of the form

$$P(B = 2md) = \frac{M - |m|}{M^2}, \qquad m = 0, \pm 1, \pm 2, \ldots, \pm(M-1) \tag{9.2–64}$$

where B denotes the noise-free received level and $2d$ is the distance between any two adjacent received signal levels.

The channel corrupts the signal transmitted through it by the addition of white Gaussian noise with zero-mean and power spectral density $\frac{1}{2}N_0$.

We assume that a symbol error occurs whenever the magnitude of the additive noise exceeds the distance d. This assumption neglects the rare event that a large noise component with magnitude exceeding d may result in a received signal level that yields a correct symbol decision. The noise component ν_m is zero-mean Gaussian with variance

$$\begin{aligned} \sigma_\nu^2 &= \tfrac{1}{2}N_0 \int_{-W}^{W} |G_R(f)|^2 df \\ &= \tfrac{1}{2}N_0 \int_{-W}^{W} |X(f)| df = \frac{2N_0}{\pi} \end{aligned} \tag{9.2–65}$$

for both the duobinary and the modified duobinary signals. Hence, an upper bound on the symbol probability of error is

$$\begin{aligned} P_e < &\sum_{m=-(M-2)}^{M-2} P(|y - 2md| > d \mid B = 2md) P(B = 2md) \\ &+ 2P[y + 2(M-1)d > d \mid B = -2(M-2)d] P[B = -2(M-1)d] \\ = &\, P(|y| > d \mid B = 0) \left\{ 2 \sum_{m=0}^{M-1} P(B = 2md) - P(B = 0) - P[B = -2(M-1)d] \right\} \\ = &\, (1 - M^{-2}) P(|y| > d \mid B = 0) \end{aligned} \tag{9.2–66}$$

But

$$\begin{aligned} P(|y| > d \mid B = 0) &= \frac{2}{\sqrt{2\pi}\sigma_\nu} \int_d^\infty e^{-x^2/2\sigma_\nu^2} dx \\ &= 2Q\left(\sqrt{\frac{\pi d^2}{2N_0}}\right) \end{aligned} \tag{9.2–67}$$

Therefore, the average probability of a symbol error is upper-bounded as

$$P_e < 2(1 - M^{-2})Q\left(\sqrt{\frac{\pi d^2}{2N_0}}\right) \tag{9.2–68}$$

The scale factor d in Equation 9.2–68 can be eliminated by expressing it in terms of the average power transmitted into the channel. For the M-ary PAM signal in which the transmitted levels are equally probable, the average power at the output of the transmitting filter is

$$P_{\text{av}} = \frac{E\left(I_m^2\right)}{T}\int_{-W}^{W}|G_T(f)|^2\,df = \frac{E\left(I_m^2\right)}{T}\int_{-W}^{W}|X(f)|df = \frac{4}{\pi T}E\left(I_m^2\right) \tag{9.2–69}$$

where $E\left(I_m^2\right)$ is the mean square value of the M signal levels, which is

$$E\left(I_m^2\right) = \tfrac{1}{3}d^2(M^2 - 1) \tag{9.2–70}$$

Therefore,

$$d^2 = \frac{3\pi P_{\text{av}}T}{4(M^2 - 1)} \tag{9.2–71}$$

By substituting the value of d^2 from Equation 9.2–71 into Equation 9.2–68, we obtain the upper bound on the symbol error probability as

$$P_e < 2\left(1 - \frac{1}{M^2}\right)Q\left(\sqrt{\left(\frac{\pi}{4}\right)^2\frac{6}{M^2 - 1}\frac{\mathcal{E}_{\text{av}}}{N_0}}\right) \tag{9.2–72}$$

where $\mathcal{E}_{\text{av}}$ is the average energy per transmitted symbol, which can be also expressed in terms of the average bit energy as $\mathcal{E}_{\text{av}} = k\mathcal{E}_{b\text{av}} = (\log_2 M)\mathcal{E}_{b\text{av}}$.

The expression in Equation 9.2–72 for the probability of error of M-ary PAM holds for both duobinary and modified duobinary partial-response signals. If we compare this result with the error probability of M-ary PAM with zero ISI, which can be obtained by using a signal pulse with a raised cosine spectrum, we note that the performance of partial-response duobinary or modified duobinary has a loss of $(\frac{1}{4}\pi)^2$, or 2.1 dB. This loss in SNR is due to the fact that the detector for the partial-response signals makes decisions on a symbol-by-symbol basis, and ignores the inherent memory contained in the received signal at its input.

Maximum-likelihood sequence detection It is clear from the above discussion that partial-response waveforms are signal waveforms with memory. This memory is conveniently represented by a trellis. For example, the trellis for the duobinary partial-response signal for binary data transmission is illustrated in Figure 9.2–12. For binary modulation, this trellis contains two states, corresponding to the two possible input values of I_m, i.e., $I_m = \pm 1$. Each branch in the trellis is labeled by two numbers. The first number on the left is the new data bit, i.e., $I_{m+1} = \pm 1$. This number determines the transition to the new state. The number on the right is the received signal level.

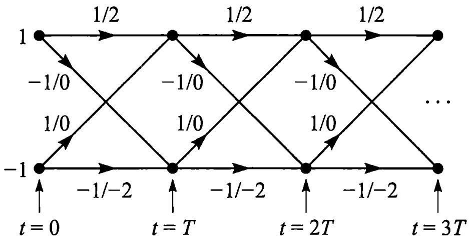


FIGURE 9.2–12
Trellis for duobinary partial-response signal.

The duobinary signal has a memory of length $L = 1$. Hence, for binary modulation the trellis has $S_t = 2$ states. In general, for M-ary modulation, the number of trellis states is M^L.

The optimum maximum-likelihood (ML) sequence detector selects the most probable path through the trellis upon observing the received data sequence $\{y_m\}$ at the sampling instants $t = mT, m = 1, 2, \ldots$. In general, each node in the trellis will have M incoming paths and M corresponding metrics. One out of the M incoming paths is selected as the most probable, based on the values of the metrics and the other $M - 1$ paths and their metrics are discarded. The surviving path at each node is then extended to M new paths, one for each of the M possible input symbols, and the search process continues. This is basically the Viterbi algorithm for performing the trellis search. Its performance is calculated in Section 9.3–4.

9.2–4 Signal Design for Channels with Distortion

In Sections 9.2–1 and 9.2–2, we described signal design criteria for the modulation filter at the transmitter and the demodulation filter at the receiver when the channel is ideal. In this section, we perform the signal design under the condition that the channel distorts the transmitted signal. We assume that the channel frequency-response $C(f)$ is known for $|f| \leq W$ and that $C(f) = 0$ for $|f| > W$. The filter responses $G_T(f)$ and $G_R(f)$ may be selected to minimize the error probability at the detector. The additive channel noise is assumed to be Gaussian with power spectral density $S_{nn}(f)$. Figure 9.2–13 illustrates the overall system under consideration.

For the signal component at the output of the demodulator, we must satisfy the condition

$$G_T(f)C(f)G_R(f) = X_d(f)e^{-j2\pi f t_0}, \qquad |f| \leq W \tag{9.2–73}$$

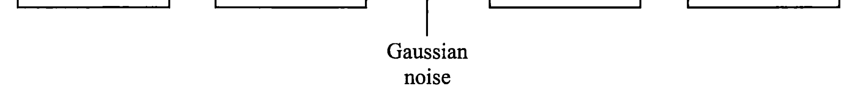


FIGURE 9.2–13
System model for the design of the modulation and demodulation filters.

where $X_d(f)$ is the desired frequency response of the cascade of the modulator, channel, and demodulator, and t_0 is a time delay that is necessary to ensure the physical realizability of the modulation and demodulation filters. The desired frequency response $X_d(f)$ may be selected to yield either zero ISI or controlled ISI at the sampling instants. We shall consider the case of zero ISI by selecting $X_d(f) = X_{rc}(f)$, where $X_{rc}(f)$ is the raised cosine spectrum with an arbitrary roll-off factor.

The noise at the output of the demodulation filter may be expressed as

$$\nu(t) = \int_{-\infty}^{\infty} n(t-\tau) g_R(\tau)\, d\tau \tag{9.2–74}$$

where $n(t)$ is the input to the filter. Since $n(t)$ is zero-mean Gaussian, $\nu(t)$ is zero-mean Gaussian, with a power spectral density

$$\mathcal{S}_{\nu\nu}(f) = \mathcal{S}_{nn}(f) |G_R(f)|^2 \tag{9.2–75}$$

For simplicity, we consider binary PAM transmission. Then, the sampled output of the matched filter is

$$y_m = x_0 I_m + \nu_m = I_m + \nu_m \tag{9.2–76}$$

where x_0 is normalized[†] to unity, $I_m = \pm d$, and ν_m represents the noise term, which is zero-mean Gaussian with variance

$$\sigma_\nu^2 = \int_{-\infty}^{\infty} \mathcal{S}_{nn}(f) |G_R(f)|^2\, df \tag{9.2–77}$$

Consequently, the probability of error is

$$P_2 = \frac{1}{\sqrt{2\pi}} \int_{d/\sigma_\nu}^{\infty} e^{-y^2/2}\, dy = Q\left(\sqrt{\frac{d^2}{\sigma_\nu^2}}\right) \tag{9.2–78}$$

The probability of error is minimized by maximizing the ratio d^2/σ_ν^2 or, equivalently, by minimizing the noise-to-signal ratio σ_ν^2/d^2.

Let us consider two possible solutions for the case in which the additive Gaussian noise is white, so that $\mathcal{S}_{nn}(f) = N_0/2$. First, suppose that we precompensate for the total channel distortion at the transmitter, so that the filter at the receiver is matched to the received signal. In this case, the transmitter and receiver filters have the magnitude characteristics

$$\begin{aligned} |G_T(f)| &= \frac{\sqrt{X_{rc}(f)}}{|C(f)|}, \qquad |f| \le W \\ |G_R(f)| &= \sqrt{X_{rc}(f)}, \qquad |f| \le W \end{aligned} \tag{9.2–79}$$

The phase characteristic of the channel frequency response $C(f)$ may also be compensated at the transmitter filter. For these filter characteristics, the average transmitted power is

---

[†] By setting $x_0 = 1$ and $I_m = \pm d$, the scaling by x_0 is incorporated into the parameter d.

$$\begin{aligned} P_{\mathrm{av}} &= \frac{E\left(I_m^2\right)}{T}\int_{-\infty}^{\infty} g_T^2(t)\,dt = \frac{d^2}{T}\int_{-W}^{W} |G_T(f)|^2\,df \\ &= \frac{d^2}{T}\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|^2}df \end{aligned} \tag{9.2–80}$$

and, hence,

$$d^2 = P_{\mathrm{av}}T\left[\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|^2}df\right]^{-1} \tag{9.2–81}$$

The noise variance at the output of the receiver filter is $\sigma_\nu^2 = N_0/2$ and, hence, the SNR at the detector is

$$\frac{d^2}{\sigma_\nu^2} = \frac{2P_{\mathrm{av}}T}{N_0}\left[\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|^2}\,df\right]^{-1} \tag{9.2–82}$$

As an alternative, suppose we split the channel compensation equally between the transmitter and receiver filters, i.e.,

$$\begin{aligned} |G_T(f)| &= \frac{\sqrt{X_{rc}(f)}}{|C(f)|^{1/2}}, \qquad |f| \le W \\ |G_R(f)| &= \frac{\sqrt{X_{rc}(f)}}{|C(f)|^{1/2}} \qquad |f| \le W \end{aligned} \tag{9.2–83}$$

The phase characteristic of $C(f)$ may also be split equally between the transmitter and receiver filters. In this case, the average transmitter power is

$$P_{av} = \frac{d^2}{T}\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|}df \tag{9.2–84}$$

and the noise variance at the output of the receiver filter is

$$\sigma_\nu^2 = \frac{N_0}{2}\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|}df \tag{9.2–85}$$

Hence, the SNR at the detector is

$$\frac{d^2}{\sigma_\nu^2} = \frac{2P_{\mathrm{av}}T}{N_0}\left[\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|}df\right]^{-2} \tag{9.2–86}$$

From Equations 9.2–82 and 9.2–86, we observe that when we express the SNR d^2/σ_ν^2 in terms of the average transmitter power P_{av}, there is a loss incurred due to channel distortion. In the case of the filters given by Equation 9.2–79, the loss is

$$10\log\int_{-W}^{W}\frac{X_{rc}(f)}{|C(f)|^2}df \tag{9.2–87}$$

and, in the case of the filters given by Equation 9.2–83, the loss is

$$10 \log \left[\int_{-W}^{W} \frac{X_{rc}(f)}{|C(f)|} df \right]^2 \tag{9.2–88}$$

We observe that when $C(f) = 1$ for $|f| \leq W$, the channel is ideal and

$$\int_{-W}^{W} X_{rc}(f)\, df = 1 \tag{9.2–89}$$

so that no loss is incurred. On the other hand, when there is amplitude distortion, $|C(f)| < 1$ for some range of frequencies in the band $|f| \leq W$ and, hence, there is a loss in SNR as given by Equations 9.2–87 and 9.2–88. The interested reader may show (see Problem 9.30) that the filters given by Equation 9.2–83 result in the smaller SNR loss.

EXAMPLE 9.2–1. Let us determine the transmitting and receiving filters given by Equation 9.2–83 for a binary communication system that transmits data at a rate of 4800 bits/s over a channel with frequency (magnitude) response

$$|C(f)| = \frac{1}{\sqrt{1 + (f/W)^2}}, \qquad |f| \leq W \tag{9.2–90}$$

where $W = 4800$ Hz. The additive noise is zero-mean white Gaussian with spectral density $\frac{1}{2}N_0 = 10^{-15}$ W/Hz.

Since $W = 1/T = 4800$, we use a signal pulse with a raised cosine spectrum and $\beta = 1$. Thus,

$$\begin{aligned} X_{rc}(f) &= \tfrac{1}{2}T[1 + \cos(\pi T|f|)] \\ &= T \cos^2\left(\frac{\pi|f|}{9600}\right) \end{aligned} \tag{9.2–91}$$

Then,

$$|G_T(f)| = |G_R(f)| = \left[1 + \left(\frac{f}{4800}\right)^2\right]^{1/4} \cos\left(\frac{\pi|f|}{9600}\right), \qquad |f| \leq 4800 \tag{9.2–92}$$

and $|G_T(f)| = |G_R(f)| = 0$, otherwise. Figure 9.2–14 illustrates the filter characteristic $G_T(f)$.

One can now use these filters to determine the amount of transmitted energy $\mathcal{E}$ required to achieve a specified error probability. This problem is left as an exercise for the reader.

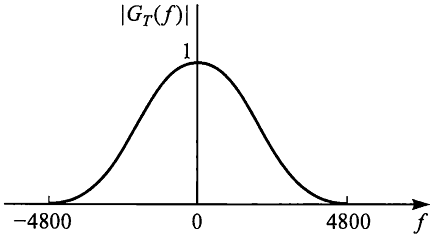


FIGURE 9.2–14
Frequency response of an optimum transmitter filter.

9.3
OPTIMUM RECEIVER FOR CHANNELS WITH ISI AND AWGN

In this section, we derive the structure of the optimum demodulator and detector for digital transmission through a nonideal band-limited channel with additive Gaussian noise. We begin with the transmitted (equivalent lowpass) signal given by Equation 9.2–1. The received (equivalent lowpass) signal is expressed as

$$r(t) = \sum_n I_n h(t - nT) + z(t) \tag{9.3–1}$$

where $h(t)$ represents the response of the channel to the input signal pulse $g(t)$ and $z(t)$ represents the additive white Gaussian noise.

First we demonstrate that the optimum demodulator can be realized as a filter matched to $h(t)$, followed by a sampler operating at the symbol rate $1/T$ and a subsequent processing algorithm for estimating the information sequence $\{I_n\}$ from the sample values. Consequently, the samples at the output of the matched filter are sufficient for the estimation of the sequence $\{I_n\}$.

9.3–1 Optimum Maximum-Likelihood Receiver

Using the Karhunen-Loève expansion, we expand the received signal $r_l(t)$ in the series

$$r_l(t) = \lim_{N\to\infty} \sum_{k=1}^{N} r_k \phi_k(t) \tag{9.3–2}$$

where $\{\phi_k(t)\}$ is a complete set of orthonormal functions and $\{r_k\}$ are the observable random variables obtained by projecting $r_l(t)$ onto the set $\{\phi_k(t)\}$. It is easily shown that

$$r_k = \sum_n I_n h_{kn} + z_k, \qquad k = 1, 2, \ldots \tag{9.3–3}$$

where h_{kn} is the value obtained from projecting $h(t - nT)$ onto $\phi_k(t)$, and z_k is the value obtained from projecting $z(t)$ onto $\phi_k(t)$. The sequence $\{z_k\}$ is Gaussian with zero-mean and covariance

$$E\left(z_k^* z_m\right) = 2N_0 \delta_{km} \tag{9.3–4}$$

The joint probability density function of the random variables $\boldsymbol{r}_N \equiv [r_1\ r_2 \cdots r_N]$ conditioned on the transmitted sequence $\boldsymbol{I}_p \equiv [I_1\ \ I_2 \cdots I_p]$, where $p \le N$, is

$$p(\boldsymbol{r}_N|\boldsymbol{I}_p) = \left(\frac{1}{2\pi N_0}\right)^N \exp\left(-\frac{1}{2N_0}\sum_{k=1}^{N}\left|r_k - \sum_n I_n h_{kn}\right|^2\right) \tag{9.3–5}$$

In the limit as the number N of observable random variables approaches infinity, the logarithm of $p(\boldsymbol{r}_N|\boldsymbol{I}_p)$ is proportional to the metrics $PM(\boldsymbol{I}_p)$, defined as

$$\begin{aligned} PM(\boldsymbol{I}_p) &= -\int_{-\infty}^{\infty}\left|r_l(t)-\sum_n I_n h(t-nT)\right|^2 dt \\ &= -\int_{-\infty}^{\infty}|r_l(t)|^2 dt + 2\mathrm{Re}\sum_n\left[I_n^*\int_{-\infty}^{\infty} r_l(t)h^*(t-nT)\,dt\right] \\ &\quad -\sum_n\sum_m I_n^* I_m\int_{-\infty}^{\infty} h^*(t-nT)h(t-mT)\,dt \end{aligned} \tag{9.3–6}$$

The maximum-likelihood estimates of the symbols $I_1, I_2, \ldots, I_p$ are those that maximize this quantity. Note, however, that the integral of $|r_l(t)|^2$ is common to all metrics, and, hence, it may be discarded. The other integral involving $r(t)$ gives rise to the variables

$$y_n \equiv y(nT) = \int_{-\infty}^{\infty} r_l(t)h^*(t-nT)\,dt \tag{9.3–7}$$

These variables can be generated by passing $r(t)$ through a filter matched to $h(t)$ and sampling the output at the symbol rate $1/T$. The samples $\{y_n\}$ form a set of sufficient statistics for the computation of $PM(\boldsymbol{I}_p)$ or, equivalently, of the correlation metrics

$$CM(\boldsymbol{I}_p) = 2\mathrm{Re}\left(\sum_n I_n^* y_n\right) - \sum_n\sum_m I_n^* I_m x_{n-m} \tag{9.3–8}$$

where, by definition, $x(t)$ is the response of the matched filter to $h(t)$ and

$$x_n \equiv x(nT) = \int_{-\infty}^{\infty} h^*(t)h(t+nT)\,dt \tag{9.3–9}$$

Hence, $x(t)$ represents the output of a filter having an impulse response $h^*(-t)$ and an excitation $h(t)$. In other words, $x(t)$ represents the autocorrelation function of $h(t)$. Consequently, $\{x_n\}$ represents the samples of the autocorrelation function of $h(t)$, taken periodically at $1/T$. We are not particularly concerned with the noncausal characteristic of the filter matched to $h(t)$, since, in practice, we can introduce a sufficiently large delay to ensure causality of the matched filter.

If we substitute for $r_l(t)$ in Equation 9.3–7 using Equation 9.3–1, we obtain

$$y_k = \sum_n I_n x_{k-n} + \nu_k \tag{9.3–10}$$

where ν_k denotes the additive noise sequence of the output of the matched filter, i.e.,

$$\nu_k = \int_{-\infty}^{\infty} z(t)h^*(t-kT)\,dt \tag{9.3–11}$$

The output of the demodulator (matched filter) at the sampling instants is corrupted by ISI as indicated by Equation 9.3–10. In any practical system, it is reasonable to assume that the ISI affects a finite number of symbols. Hence, we may assume that $x_n = 0$ for $|n| > L$. Consequently, the ISI observed at the output of the demodulator may be viewed as the output of a finite state machine. This implies that the channel output with ISI may be represented by a trellis diagram, and the maximum-likelihood

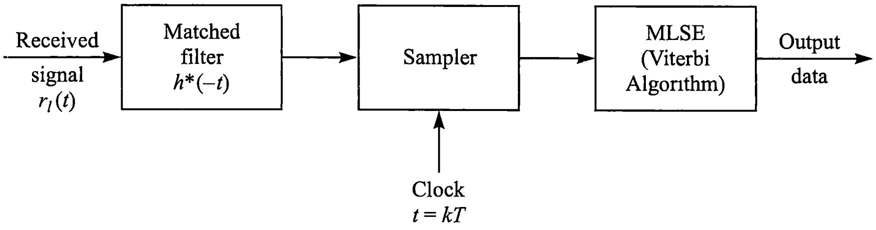


FIGURE 9.3–1
Optimum receiver for an AWGN channel with ISI.

estimate of the information sequence $(I_1, I_2, \ldots, I_p)$ is simply the most probable path through the trellis given the received demodulator output sequence $\{y_n\}$. Clearly, the Viterbi algorithm provides an efficient means for performing the trellis search.

The metrics that are computed for the MLSE of the sequence $\{I_k\}$ are given by Equation 9.3–8. It can be seen that these metrics can be computed recursively in the Viterbi algorithm, according to the relation

$$CM_n(\boldsymbol{I}_n) = CM_{n-1}(\boldsymbol{I}_{n-1}) + \text{Re}\left[I_n^*\left(2y_n - x_0 I_n - 2\sum_{m=1}^{L} x_m I_{n-m}\right)\right] \tag{9.3–12}$$

Figure 9.3–1 illustrates the block diagram of the optimum receiver for an AWGN channel with ISI.

9.3–2 A Discrete-Time Model for a Channel with ISI

In dealing with band-limited channels that result in ISI, it is convenient to develop an equivalent discrete-time model for the analog (continuous-time) system. Since the transmitter sends discrete-time symbols at a rate of $1/T$ symbols/s and the sampled output of the matched filter at the receiver is also a discrete-time signal with samples occurring at a rate of $1/T$ per second, it follows that the cascade of the analog filter at the transmitter with impulse response $g(t)$, the channel with impulse response $c(t)$, the matched filter at the receiver with impulse response $h^*(-t)$, and the sampler can be represented by an equivalent discrete-time tranversal filter having tap gain coefficients $\{x_k\}$. Consequently, we have an equivalent discrete-time transversal filter that spans a time interval of $2LT$ seconds. Its input is the sequence of information symbols $\{I_k\}$ and its output is the discrete-time sequence $\{y_k\}$ given by Equation 9.3–10. The equivalent discrete-time model is shown in Figure 9.3–2.

The major difficulty with this discrete-time model occurs in the evaluation of performance of the various equalization or estimation techniques that are discussed in the following sections. The difficulty is caused by the correlations in the noise sequence $\{\nu_k\}$ at the output of the matched filter. That is, the set of noise variables $\{\nu_k\}$ is a Gaussian-distributed sequence with zero-mean and autocorrelation function (see Problem 9.36)

$$E\left(\nu_k^* \nu_j\right) = \begin{cases} 2N_0 x_{j-k} & (|k-j| \le L) \\ 0 & (\text{otherwise}) \end{cases} \tag{9.3–13}$$

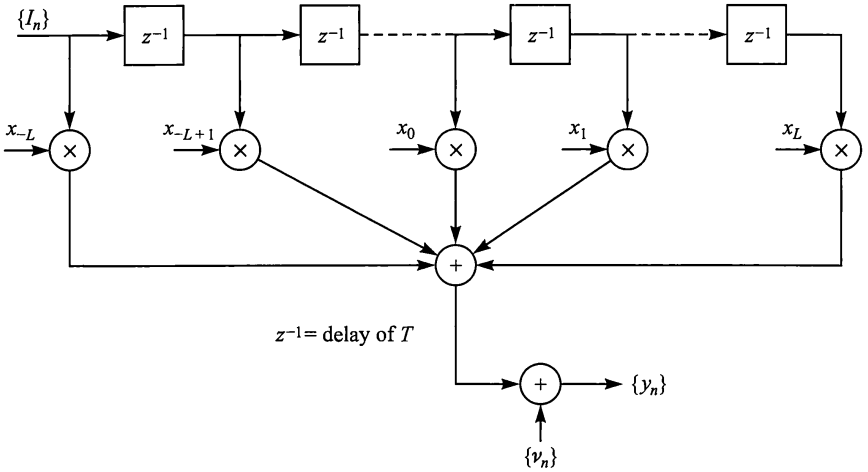


FIGURE 9.3–2
Equivalent discrete-time model of channel with intersymbol interference.

Hence, the noise sequence is correlated unless $x_k = 0, k \neq 0$. Since it is more convenient to deal with the white noise sequence when calculating the error rate performance, it is desirable to whiten the noise sequence by further filtering the sequence $\{y_k\}$. A discrete-time noise-whitening filter is determined as follows.

Let $X(z)$ denote the (two-sided) z transform of the sampled autocorrelation function $\{x_k\}$, i.e.,

$$X(z) = \sum_{k=-L}^{L} x_k z^{-k} \tag{9.3–14}$$

Since $x_k = x^*_{-k}$, it follows that $X(z) = X^*(1/z^*)$ and the $2L$ roots of $X(z)$ have the symmetry that if ρ is a root, $1/\rho^*$ is also a root. Hence, $X(z)$ can be factored and expressed as

$$X(z) = F(z)F^*\left(\frac{1}{z^*}\right) \tag{9.3–15}$$

where $F(z)$ is a polynomial of degree L having the roots $\rho_1, \rho_2, \ldots, \rho_L$ and $F^*(1/z^*)$ is a polynomial of degree L having the roots $1/\rho_1^*, 1/\rho_2^*, \ldots, 1/\rho_L^*$. Assuming that there are no roots on the unit circle, an appropriate noise-whitening filter has a z transform $1/F^*(1/z^*)$. Since there are 2^L possible choices for the roots of $F^*(1/z^*)$, each choice resulting in a filter characteristic that is identical in magnitude but different in phase from other choices of the roots, we propose to choose the unique $F^*(1/z^*)$ that results in an anticausal impulse response with poles corresponding to the zeros of $X(z)$ that are outside the unit circle. Such an anticausal filter is stable. Selecting the noise-whitening filter in this manner ensures that the resulting channel response, characterized by $F(z)$, is minimum phase. Consequently, passage of the sequence $\{y_k\}$ through the digital filter $1/F^*(1/z^*)$ results in an output sequence $\{v_k\}$ that can be expressed as

$$v_k = \sum_{n=0}^{L} f_n I_{k-n} + \eta_k \tag{9.3–16}$$

where $\{\eta_k\}$ is a white Gaussian noise sequence and $\{f_k\}$ is a set of tap coefficients of an equivalent discrete-time transversal filter having a transfer function $F(z)$. The cascade of the matched filter, the sampler, and the noise-whitening filter is called the *whitened matched filter* (WMF).

It is convenient to normalize the energy of $F(z)$ to unity, i.e.,

$$\sum_{n=0}^{L} |f_n|^2 = 1$$

The minimum-phase condition on $F(z)$ implies that the energy in the first M values of the impulse response $\{f_0, f_1, \ldots, f_M\}$ is a maximum for every M.

In summary, the cascade of the transmitting filter $g(t)$, the channel $c(t)$, the matched filter $h^*(-t)$, the sampler, and the discrete-time noise-whitening filter $1/F^*(1/z^*)$ can be represented as an equivalent discrete-time transversal filter having the set $\{f_k\}$ as its tap coefficients. The additive noise sequence $\{\eta_k\}$ corrupting the output of the discrete-time transversal filter is a white Gaussian noise sequence having zero-mean and variance N_0. Figure 9.3–3 illustrates the model of the equivalent discrete-time system with white noise. We refer to this model as the *equivalent discrete-time white noise filter model.*

EXAMPLE 9.3–1. Suppose that the transmitter signal pulse $g(t)$ has duration T and unit energy and the received signal pulse is $h(t) = g(t) + ag(t - T)$. Let us determine the equivalent discrete-time white noise filter model. The sampled autocorrelation function is given by

$$x_k = \begin{cases} a^* & (k = -1) \\ 1 + |a|^2 & (k = 0) \\ a & (k = 1) \end{cases} \tag{9.3–17}$$

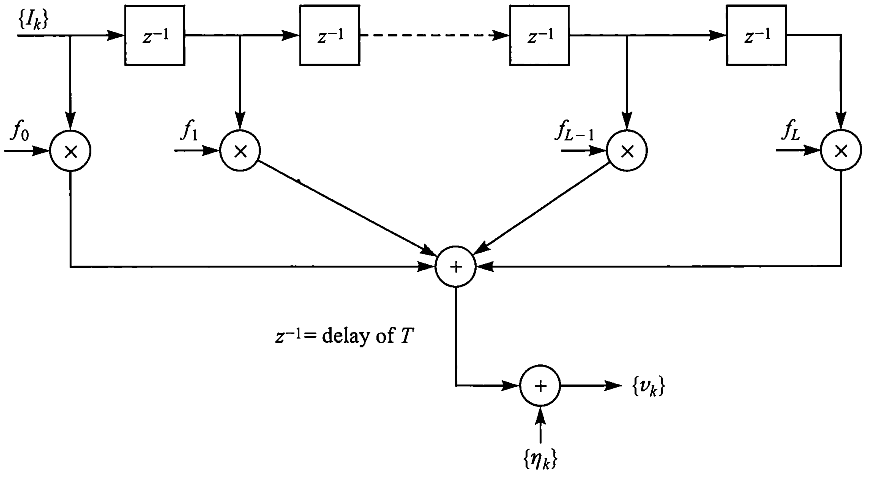


FIGURE 9.3–3
Equivalent discrete-time model of intersymbol interference channel with AWGN.

The z transform of x_k is

$$\begin{aligned} X(z) &= \sum_{k=-1}^{1} x_k z^{-k} \\ &= a^* z + (1 + |a|^2) + a z^{-1} \\ &= (a z^{-1} + 1)(a^* z + 1) \end{aligned} \tag{9.3–18}$$

Under the assumption that $|a| < 1$, one chooses $F(z) = az^{-1} + 1$, so that the equivalent transversal filter consists of two taps having tap gain coefficients $f_0 = 1$, $f_1 = a$. Note that the correlation sequence $\{x_k\}$ may be expressed in terms of the $\{f_n\}$ as

$$x_k = \sum_{n=0}^{L-k} f_n^* f_{n+k}, \qquad k = 0, 1, 2, \ldots, L \tag{9.3–19}$$

When the channel impulse response is changing slowly with time, the matched filter at the receiver becomes a time-variable filter. In this case, the time variations of the channel/matched-filter pair result in a discrete-time filter with time-variable coefficients. As a consequence, we have time-variable intersymbol interference effects, which can be modeled by the filter illustrated in Figure 9.3–3, where the tap coefficients are slowly varying with time.

The discrete-time white noise linear filter model for the intersymbol interference effects that arise in high-speed digital transmission over nonideal band-limited channels will be used throughout the remainder of this chapter in our discussion of compensation techniques for the interference. In general, the compensation methods are called *equalization techniques* or *equalization algorithms*.

9.3–3 Maximum-Likelihood Sequence Estimation (MLSE) for the Discrete-Time White Noise Filter Model

In the presence of intersymbol interference that spans $L + 1$ symbols (L interfering components), the MLSE criterion is equivalent to the problem of estimating the state of a discrete-time finite-state machine. The finite-state machine in this case is the equivalent discrete-time channel with coefficients $\{f_k\}$, and its state at any instant in time is given by the L most recent inputs, i.e., the state at time k is

$$S_k = (I_{k-1}, I_{k-2}, \ldots, I_{k-L}) \tag{9.3–20}$$

where $I_k = 0$ for $k \leq 0$. Hence, if the information symbols are M-ary, the channel filter has M^L states. Consequently, the channel is described by an M^L-state trellis and the Viterbi algorithm may be used to determine the most probable path through the trellis.

The metrics used in the trellis search are akin to the metrics used in soft-decision decoding of convolutional codes. In brief, we begin with the samples $v_1, v_2, \ldots, v_{L+1}$, from which we compute the M^{L+1} metrics

$$\sum_{k=1}^{L+1} \ln p(v_k | I_k, I_{k-1}, \ldots, I_{k-L}) \tag{9.3–21}$$

The M^{L+1} possible sequences of $I_{L+1}, I_L, \ldots, I_2, I_1$ are subdivided into M^L groups corresponding to the M^L states $(I_{L+1}, I_L, \ldots, I_2)$. Note that the M sequences in each group (state) differ in I_1 and correspond to the paths through the trellis that merge at a single node. From the M sequences in each of the M^L states, we select the sequence with the largest probability (with respect to I_1) and assign to the surviving sequence the metric

$$\begin{aligned} PM_1(\boldsymbol{I}_{L+1}) &\equiv PM_1(I_{L+1}, I_L, \ldots, I_2) \\ &= \max_{I_1} \sum_{k=1}^{L+1} \ln p(v_k | I_k, I_{k-1}, \ldots, I_{k-L}) \end{aligned} \tag{9.3–22}$$

The $M - 1$ remaining sequences from each of the M^L groups are discarded. Thus, we are left with M^L surviving sequences and their metrics.

Upon reception of v_{L+2}, the M^L surviving sequences are extended by one stage, and the corresponding M^{L+1} probabilities for the extended sequences are computed using the previous metrics and the new increment, which is $\ln p(v_{L+2}|I_{L+2}, I_{L+1}, \ldots, I_2)$. Again, the M^{L+1} sequences are subdivided into M^L groups corresponding to the M^L possible states $(I_{L+2}, \ldots, I_3)$ and the most probable sequence from each group is selected, while the other $M - 1$ sequences are discarded.

The procedure described continues with the reception of subsequent signal samples. In general, upon reception of v_{L+k}, the metrics[†]

$$PM_k(\boldsymbol{I}_{L+k}) = \max_{I_k}[\ln p(v_{L+k}|I_{L+k}, \ldots, I_k) + PM_{k-1}(\boldsymbol{I}_{L+k-1}))] \tag{9.3–23}$$

that are computed give the probabilities of the M^L surviving sequences. Thus, as each signal sample is received, the Viterbi algorithm involves first the computation of the M^{L+1} probabilities

$$\ln p(v_{L+k}|I_{L+k}, \ldots, I_k) + PM_{k-1}(\boldsymbol{I}_{L+k-1}) \tag{9.3–24}$$

corresponding to the M^{L+1} sequences that form the continuations of the M^L surviving sequences from the previous stage of the process. Then the M^{L+1} sequences are subdivided into M^L groups, with each group containing M sequences that terminate in the same set of symbols $I_{L+k}, \ldots, I_{k+1}$ and differ in the symbol I_k. From each group of M sequences, we select the one having the largest probability as indicated by Equation 9.3–23, while the remaining $M - 1$ sequences are discarded. Thus, we are left again with M^L sequences having the metrics $PM_k(\boldsymbol{I}_{L+k})$.

As indicated previously, the delay in detecting each information symbol is variable. In practice, the variable delay is avoided by truncating the surviving sequences to the q most recent symbols, where $q \gg L$, thus achieving a fixed delay. In the case that the M^L surviving sequences at time k disagree on the symbol I_{k-q}, the symbol in the most probable sequence may be chosen. The loss of performance resulting from this suboptimum decision procedure is negligible if $q \geq 5L$.

EXAMPLE 9.3–2. For illustrative purposes, suppose that a duobinary signal pulse is employed to transmit four-level ($M = 4$) PAM. Thus, each symbol is a number selected from the set $\{-3, -1, 1, 3\}$. The controlled intersymbol interference in this partial-response signal is represented by the equivalent discrete-time channel model shown in

---

[†] We observe that the metrics $PM_k(\mathrm{I})$ are simply related to the Euclidean distance metrics $DM_k(\mathrm{I})$ when the additive noise is Gaussian.

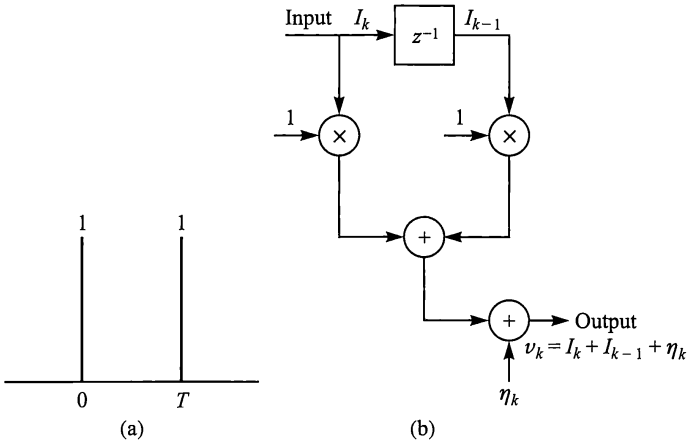


FIGURE 9.3–4

Equivalent discrete-time model for intersymbol interference resulting from a duobinary pulse.

Figure 9.3–4. Suppose we have received v_1 and v_2, where

$$\begin{aligned} v_1 &= I_1 + \eta_1 \\ v_2 &= I_2 + I_1 + \eta_2 \end{aligned} \tag{9.3–25}$$

and $\{\eta_i\}$ is a sequence of statistically independent zero-mean Gaussian noise. We may now compute the 16 metrics

$$PM_1(I_2, I_1) = -\sum_{k=1}^{2}\left(v_k - \sum_{j=0}^{1} I_{k-j}\right)^2, \qquad I_1, I_2 = \pm 1, \pm 3 \tag{9.3–26}$$

where $I_k = 0$ for $k \leq 0$.

Note that any subsequently received signals $\{v_i\}$ do not involve I_1. Hence, at this stage, we may discard 12 of the 16 possible pairs $\{I_1, I_2\}$. This step is illustrated by the tree diagram shown in Figure 9.3–5. In other words, after computing the 16 metrics corresponding to the 16 paths in the tree diagram, we discard three out of the four paths that terminate with $I_2 = 3$ and save the most probable of these four. Thus, the metric for the surviving path is

$$PM_1(I_2 = 3, I_1) = \max_{I_1}\left[-\sum_{k=1}^{2}\left(v_k - \sum_{j=0}^{1} I_{k-j}\right)^2\right]$$

The process is repeated for each set of four paths terminating with $I_2 = 1$, $I_2 = -1$, and $I_2 = -3$. Thus four paths and their corresponding metrics survive after v_1 and v_2 are received.

When v_3 is received, the four paths are extended as shown in Figure 9.3–5 to yield 16 paths and 16 corresponding metrics given by

$$PM_2(I_3, I_2, I_1) = PM_1(I_2, I_1) - \left(v_3 - \sum_{j=0}^{1} I_{3-j}\right)^2 \tag{9.3–27}$$

Of the four paths terminating with the $I_3 = 3$, we save the most probable. This procedure is again repeated for $I_3 = 1$, $I_3 = -1$, and $I_3 = -3$. Consequently, only four paths

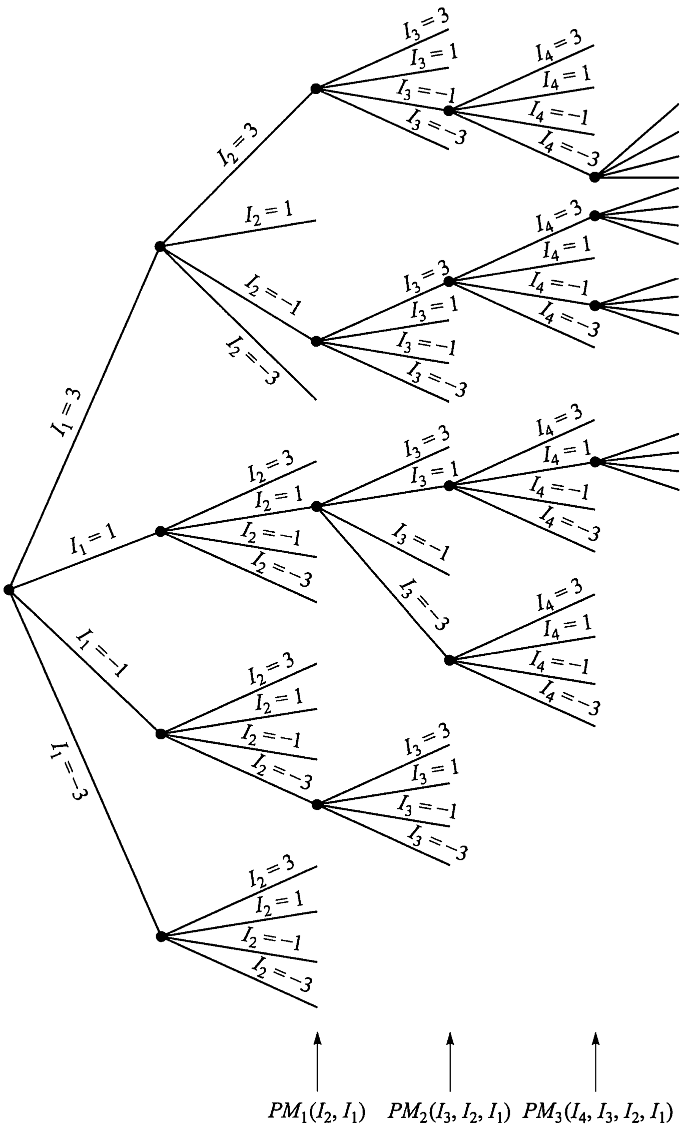


FIGURE 9.3–5
Tree diagram for Viterbi decoding of the duobinary pulse.

survive at this stage. The procedure is then repeated for each subsequently received signal v_k for $k > 3$.

9.3–4 Performance of MLSE for Channels with ISI

We shall now determine the probability of error for the MLSE of the received information sequence when the information is transmitted via PAM and the additive noise is Gaussian. The similarity between a convolutional code and a finite-duration intersymbol interference channel implies that the method for computing the error probability for the latter carries over from the former. In particular, the method for computing the performance of soft-decision decoding of a convolutional code by means of the Viterbi algorithm, described in Section 8.3, applies with some modification.

In PAM signaling with the additive Gaussian noise and intersymbol interference, the metrics used in the Viterbi algorithm may be expressed as in Equation 9.3–23, or, equivalently, as

$$PM_{k-L}(\boldsymbol{I}_k) = PM_{k-L-1}(\boldsymbol{I}_{k-1}) - \left(v_k - \sum_{j=0}^{L} f_j I_{k-j}\right)^2 \tag{9.3–28}$$

where the symbols $\{I_n\}$ may take the values $\pm d, \pm 3d, \ldots, \pm(M-1)d$, and $2d$ is the distance between successive levels. The trellis has M^L states, defined at time k as

$$S_k = (I_{k-1}, I_{k-2}, \ldots, I_{k-L}) \tag{9.3–29}$$

Let the estimated symbols from the Viterbi algorithm be denoted by $\{\tilde{I}_n\}$ and the corresponding estimated state at time k by

$$\tilde{S}_k = (\tilde{I}_{k-1}, \tilde{I}_{k-2}, \ldots, \tilde{I}_{k-L}) \tag{9.3–30}$$

Now suppose that the estimated path through the trellis diverges from the correct path at time k and remerges with the correct path at time $k+l$. Thus, $\tilde{S}_k = S_k$ and $\tilde{S}_{k+1} = S_{k+1}$, but $\tilde{S}_m \neq S_m$ for $k < m < k+l$. As in a convolutional code, we call this an *error event*. Since the channel spans $L+1$ symbols, it follows that $l \geq L+1$.

For such an error event, we have $\tilde{I}_k \neq I_k$ and $\tilde{I}_{k+l-L-1} \neq I_{k+l-L-1}$, but $\tilde{I}_m = I_m$ for $k - L \leq m \leq k-1$ and $k+l-L \leq m \leq k+l-1$. It is convenient to define an error vector $\boldsymbol{\varepsilon}$ corresponding to this error event as

$$\boldsymbol{\varepsilon} = [\varepsilon_k \quad \varepsilon_{k+1} \quad \cdots \quad \varepsilon_{k+l-L-1}] \tag{9.3–31}$$

where the components of $\boldsymbol{\varepsilon}$ are defined as

$$\varepsilon_j = \frac{1}{2d}(I_j - \tilde{I}_j), \qquad j = k, k+1, \ldots, k+l-L-1 \tag{9.3–32}$$

The normalization factor of $2d$ in Equation 9.3–32 results in elements ε_j that take on the values $0, \pm 1, \pm 2, \pm 3, \ldots, \pm(M-1)$. Moreover, the error vector is characterized by the properties that $\varepsilon_k \neq 0$, $\varepsilon_{k+l-L-1} \neq 0$, and there is no sequence of L consecutive elements that are zero. Associated with the error vector in Equation 9.3–31 is the polynomial of degree $l-L-1$,

$$\varepsilon(z) = \varepsilon_k + \varepsilon_{k+1} z^{-1} + \varepsilon_{k+2} z^{-2} + \cdots + \varepsilon_{k+l-L-1} z^{-(l-L-1)} \tag{9.3–33}$$

We wish to determine the probability of occurrence of the error event that begins at time k and is characterized by the error vector $\boldsymbol{\varepsilon}$ given in Equation 9.3–31 or, equivalently, by the polymonial given in Equation 9.3–33. To accomplish this, we follow the procedure developed by Forney (1972). Specifically, for the error event $\boldsymbol{\varepsilon}$ to occur, the following three subevents E_1, E_2, and E_3 must occur:

E_1: At time k, $\tilde{S}_k = S_k$.

E_2: The information symbols $I_k, I_{k+1}, \ldots, I_{k+l-L-1}$ when added to the scaled error sequence $2d(\varepsilon_k, \varepsilon_{k+1}, \ldots, \varepsilon_{k+l-L-1})$ must result in an allowable sequence, i.e., the sequence $\tilde{I}_k, \tilde{I}_{k+1}, \ldots, \tilde{I}_{k+l-L-1}$ must have values selected from $\pm d, \pm 3d, \pm \cdots \pm (M-1)d$.

E_3: For $k \leq m < k+l$, the sum of the branch metrics of the estimated path exceeds the sum of the branch metrics of the correct path.

The probability of occurrence of E_3 is

$$P(E_3) = P\left[\sum_{i=k}^{k+l-1}\left(v_i - \sum_{j=0}^{L} f_j \tilde{I}_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\left(v_i - \sum_{j=0}^{L} f_j I_{i-j}\right)^2\right] \tag{9.3–34}$$

But

$$v_i = \sum_{j=0}^{L} f_j I_{i-j} + \eta_i \tag{9.3–35}$$

where $\{\eta_i\}$ is a real-valued white Gaussian noise sequence. Substitution of Equation 9.3–35 into Equation 9.3–34 yields

$$\begin{aligned} P(E_3) &= P\left[\sum_{i=k}^{k+l-1}\left(\eta_i + 2d\sum_{j=0}^{L} f_j \varepsilon_{i-j}\right)^2 < \sum_{i=k}^{k+l-1}\eta_i^2\right] \\ &= P\left[4d\sum_{i=k}^{k+l-1}\eta_i\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right) < -4d^2\sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right)^2\right] \end{aligned} \tag{9.3–36}$$

where $\varepsilon_j = 0$ for $j < k$ and $j > k + l - L - 1$. If we define

$$\alpha_i = \sum_{j=0}^{L} f_j \varepsilon_{i-j} \tag{9.3–37}$$

then Equation 9.3–36 may be expressed as

$$P(E_3) = P\left(\sum_{i=k}^{k+l-1}\alpha_i\eta_i < -d\sum_{i=k}^{k+1-1}\alpha_i^2\right) \tag{9.3–38}$$

where the factor of $4d$ common to both terms has been dropped. Now Equation 9.3–38 is just the probability that a linear combination of statistically independent Gaussian random variables is less than some negative number. Thus

$$P(E_3) = Q\left(\sqrt{\frac{2d^2}{N_0}\sum_{i=k}^{k+l-1}\alpha_i^2}\right) \tag{9.3–39}$$

For convenience, we define

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{i=k}^{k+l-1}\alpha_i^2 = \sum_{i=k}^{k+l-1}\left(\sum_{j=0}^{L} f_j\varepsilon_{i-j}\right)^2 \tag{9.3–40}$$

where $\varepsilon_j = 0$ for $j < k$ and $j > k + l - L - 1$. Note that the $\{\alpha_i\}$ resulting from the convolution of $\{f_i\}$ with $\{\varepsilon_j\}$ are the coefficients of the polynomial

$$\begin{aligned} \alpha(z) &= F(z)\varepsilon(z) \\ &= \alpha_k + \alpha_{k+1}z^{-1} + \cdots + \alpha_{k+l-1}z^{-(l-1)} \end{aligned} \tag{9.3–41}$$

Furthermore, $\delta^2(\boldsymbol{\varepsilon})$ is simply equal to the coefficient of z^0 in the polynomial

$$\begin{aligned}\alpha(z)\alpha(z^{-1}) &= F(z)F(z^{-1})\varepsilon(z)\varepsilon(z^{-1})\\ &= X(z)\varepsilon(z)\varepsilon(z^{-1})\end{aligned} \tag{9.3–42}$$

We call $\delta^2(\boldsymbol{\varepsilon})$ the *Euclidean weight* of the error event ε.

An alternative method for representing the result of convolving $\{f_j\}$ with $\{\varepsilon_j\}$ is the matrix form

$$\boldsymbol{\alpha} = \boldsymbol{e}\boldsymbol{f}$$

where $\boldsymbol{\alpha}$ is an l-dimensional vector, $\boldsymbol{f}$ is an $(L+1)$-dimensional vector, and $\mathbf{e}$ is an $l \times (L+1)$ matrix defined as

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_k \\ a_{k+1} \\ \vdots \\ \alpha_{k+l-1} \end{bmatrix}, \qquad \boldsymbol{f} = \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_L \end{bmatrix}$$

$$\boldsymbol{e} = \begin{bmatrix} \varepsilon_k & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \varepsilon_{k+1} & \varepsilon_k & 0 & \cdots & 0 & \cdots & 0 \\ \varepsilon_{k+2} & \varepsilon_{k+1} & \varepsilon_k & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ \varepsilon_{k+l-1} & \cdots & \cdots & \cdots & \cdots & \cdots & \varepsilon_{k+l-L-1} \end{bmatrix} \tag{9.3–43}$$

Then

$$\begin{aligned}\delta^2(\boldsymbol{\varepsilon}) &= \boldsymbol{\alpha}^t\boldsymbol{\alpha}\\ &= \boldsymbol{f}^t\boldsymbol{e}^t\boldsymbol{e}\boldsymbol{f}\\ &= \boldsymbol{f}^t\boldsymbol{A}\boldsymbol{f}\end{aligned} \tag{9.3–44}$$

where $\boldsymbol{A}$ is an $(L+1) \times (L+1)$ matrix of the form

$$\boldsymbol{A} = \boldsymbol{e}^t\boldsymbol{e} = \begin{bmatrix} \beta_0 & \beta_1 & \beta_2 & \cdots & \beta_L \\ \beta_1 & \beta_0 & \beta_1 & \cdots & \beta_{L-1} \\ \beta_2 & \beta_1 & \beta_0 & \beta_1 & \beta_{L-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_L & \cdots & \cdots & \cdots & \beta_0 \end{bmatrix} \tag{9.3–45}$$

and

$$\beta_m = \sum_{i=k}^{k+l-1-m} \varepsilon_i \varepsilon_{i+m} \tag{9.3–46}$$

We may use either Equations 9.3–40 and 9.3–41 or Equations 9.3–45 and 9.3–46 in evaluating the error rate performance. We consider these computations later. For now we conclude that the probability of the subevent E_3, given by Equations 9.3–39, may be expressed as

$$\begin{aligned} P(E_3) &= Q\left(\sqrt{\frac{2d^2}{N_0}\delta^2(\boldsymbol{\varepsilon})}\right) \\ &= Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2(\boldsymbol{\varepsilon})}\right) \end{aligned} \tag{9.3–47}$$

where we have used the relation

$$d^2 = \frac{3}{M^2-1}TP_{\text{av}} \tag{9.3–48}$$

to eliminate d^2 and $\gamma_{\text{av}} = TP_{\text{av}}/N_0$. Note that, in the absence of intersymbol interference, $\delta^2(\boldsymbol{\varepsilon}) = 1$ and $P(E_3)$ is proportional to the symbol error probability of M-ary PAM.

The probability of the subevent E_2 depends only on the statistical properties of the input sequence. We assume that the information symbols are equally probable and that the symbols in the transmitted sequence are statistically independent. Then, for an error of the form $|\varepsilon_i| = j$, $j = 1, 2, \ldots, M-1$, there are $M - j$ possible values of I_i such that

$$I_i = \tilde{I}_i + 2d\varepsilon_i$$

Hence

$$P(E_2) = \prod_{i=0}^{l-L-1} \frac{M - |\varepsilon_i|}{M} \tag{9.3–49}$$

The probability of the subevent E_1 is much more difficult to compute exactly because of its dependence on the subevent E_3. That is, we must compute $P(E_1|E_3)$. However, $P(E_1|E_3) = 1 - P_e$, where P_e is the symbol error probability. Hence $P(E_1|E_3)$ is well approximated (and upper-bounded) by unity for reasonably low symbol error probabilities. Therefore, the probability of the error event $\boldsymbol{\varepsilon}$ is well approximated and upper-bounded as

$$P(\boldsymbol{\varepsilon}) \le Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2(\boldsymbol{\varepsilon})}\right) \prod_{i=0}^{l-L-1} \frac{M - |\varepsilon_i|}{M} \tag{9.3–50}$$

Let E be the set of all error events $\boldsymbol{\varepsilon}$ starting at time k and let $w(\boldsymbol{\varepsilon})$ be the corresponding number of nonzero components (Hamming weight or number of symbol errors) in each error event $\boldsymbol{\varepsilon}$. Then the probability of a symbol error is upper-bounded (union bound) as

$$P_e \le \sum_{\boldsymbol{\varepsilon} \in E} w(\boldsymbol{\varepsilon}) P(\boldsymbol{\varepsilon}) \le \sum_{\boldsymbol{\varepsilon} \in E} w(\boldsymbol{\varepsilon}) Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2(\boldsymbol{\varepsilon})}\right) \prod_{i=0}^{l-L-1} \frac{M-|\varepsilon_i|}{M} \tag{9.3–51}$$

Now let D be the set of all $\delta(\boldsymbol{\varepsilon})$. For each $\delta \in D$, let E_δ be the subset of error events for which $\delta(\boldsymbol{\varepsilon}) = \delta$. Then Equation 9.3–51 may be expressed as

$$\begin{aligned} P_e &\le \sum_{\delta \in D} Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2}\right) \left[\sum_{\boldsymbol{\varepsilon} \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M-|\varepsilon_i|}{M}\right] \\ &\le \sum_{\delta \in D} K_\delta Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2}\right) \end{aligned} \tag{9.3–52}$$

where

$$K_\delta = \sum_{\boldsymbol{\varepsilon} \in E_\delta} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M-|\varepsilon_i|}{M} \tag{9.3–53}$$

The expression for the error probability in Equation 9.3–52 is similar to the form of the error probability for a convolutional code with soft-decision decoding given by Equation 8.2–19. The weighting factors $\{K_\delta\}$ may be determined by means of the error state diagram, which is akin to the state diagram of a convolutional encoder. This approach has been illustrated by Forney (1972) and Viterbi and Omura (1979).

In general, however, the use of the error state diagram for computing P_e is tedious. Instead, we may simplify the computation of P_e by focusing on the dominant term in the summation of Equation 9.3–52. Because of the exponential dependence of each term in the sum, the expression P_e is dominated by the term corresponding to the minimum value of δ, denoted as $\delta_{\min}$. Hence the symbol error probability may be approximated as

$$P_e \approx K_{\delta_{\min}} Q\left(\sqrt{\frac{6}{M^2-1}\gamma_{\text{av}}\delta^2_{\min}}\right) \tag{9.3–54}$$

where

$$K_{\delta_{\min}} = \sum_{\boldsymbol{\varepsilon} \in E_{\delta_{\min}}} w(\boldsymbol{\varepsilon}) \prod_{i=0}^{l-L-1} \frac{M-|\varepsilon_i|}{M} \tag{9.3–55}$$

In general, $\delta^2_{\min} \le 1$. Hence, $10 \log \delta^2_{\min}$ represents the loss in SNR due to intersymbol interference.

The minimum value of δ may be determined either from Equation 9.3–40 or from evaluation of the quadratic form in Equation 9.3–44 for different error sequences. In the following two examples we use Equation 9.3–40.

EXAMPLE 9.3–3. Consider a two path channel ($L = 1$) with arbitrary coefficients f_0 and f_1 satisfying the constraint $f_0^2 + f_1^2 = 1$. The channel characteristic is

$$F(z) = f_0 + f_1 z^{-1} \tag{9.3–56}$$

For an error event of length n,

$$\varepsilon(z) = \varepsilon_0 + \varepsilon_1 z^{-1} + \cdots + \varepsilon_{n-1} z^{-(n-1)}, \qquad n \geq 1 \tag{9.3–57}$$

The product $\alpha(z) = F(z)\varepsilon(z)$ may be expressed as

$$\alpha(z) = \alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n} \tag{9.3–58}$$

where $\alpha_0 = \varepsilon_0 f_0$ and $\alpha_n = f_1 \varepsilon_{n-1}$. Since $\varepsilon_0 \neq 0$, $\varepsilon_{n-1} \neq 0$, and

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{k=0}^{n} \alpha_k^2 \tag{9.3–59}$$

it follows that

$$\delta_{\min}^2 \geq f_0^2 + f_1^2 = 1$$

Indeed, $\delta_{\min}^2 = 1$ when a single error occurs, i.e., $\varepsilon(z) = \varepsilon_0$. Thus, we conclude that there is no loss in SNR in maximum-likelihood sequence estimation of the information symbols when the channel dispersion has length 2.

EXAMPLE 9.3–4. The controlled intersymbol interference in a partial-response signal may be viewed as having been generated by a time-dispersive channel. Thus, the intersymbol interference from a duobinary pulse may be represented by the (normalized) channel characteristic

$$F(z) = \sqrt{\tfrac{1}{2}} + \sqrt{\tfrac{1}{2}} z^{-1} \tag{9.3–60}$$

Similarly, the representation for a modified duobinary pulse is

$$F(z) = \sqrt{\tfrac{1}{2}} - \sqrt{\tfrac{1}{2}} z^{-2} \tag{9.3–61}$$

The minimum distance $\delta_{\min}^2 = 1$ for any error event of the form

$$\varepsilon(z) = \pm(1 - z^{-1} - z^{-2} \cdots - z^{-(n-1)}), \qquad n \geq 1 \tag{9.3–62}$$

for the channel given by Equation 9.3–60, since

$$\alpha(z) = \pm\sqrt{\tfrac{1}{2}} \mp \sqrt{\tfrac{1}{2}} z^{-n}$$

Similarly, when

$$\varepsilon(z) = \pm(1 + z^{-2} + z^{-4} + \cdots + z^{-2(n-1)}), \qquad n \geq 1 \tag{9.3–63}$$

$\delta_{\min}^2 = 1$ for the channel given by Equation 9.3–61 since

$$\alpha(z) = \pm\sqrt{\tfrac{1}{2}} \mp \sqrt{\tfrac{1}{2}} z^{-2n}$$

Hence the MLSE of these two partial-response signals result in no loss in SNR. In contrast, the suboptimum symbol-by-symbol detection described previously resulted in a 2.1-dB loss.

The constant $K_{\delta_{\min}}$ is easily evaluated for these two signals. With precoding, the number of output symbol errors (Hamming weight) associated with the error events in Equations 9.3–62 and 9.3–63 is two. Hence,

$$K_{\delta_{\min}} = 2\sum_{n=1}^{\infty}\left(\frac{M-1}{M}\right)^n = 2(M-1) \tag{9.3–64}$$

On the other hand, without precoding, these error events result in n symbol errors, and, hence,

$$K_{\delta_{\min}} = 2\sum_{n=1}^{\infty} n\left(\frac{M-1}{M}\right)^n = 2M(M-1) \tag{9.3–65}$$

As a final exercise, we consider the evaluation of $\delta^2_{\min}$ from the quadratic form in Equation 9.3–44. The matrix A of the quadratic form is positive-definite; hence, all its eigenvalues are positive. If $\{\mu_k(\boldsymbol{\varepsilon})\}$ are the eigenvalues and $\{\boldsymbol{v}_k(\boldsymbol{\varepsilon})\}$ are the corresponding orthonormal eigenvectors of A for an error event $\boldsymbol{\varepsilon}$, then the quadratic form in Equation 9.3–44 can be expressed as

$$\delta^2(\boldsymbol{\varepsilon}) = \sum_{k=1}^{L+1} \mu_k(\boldsymbol{\varepsilon})[\boldsymbol{f}^t \boldsymbol{v}_k(\boldsymbol{\varepsilon})]^2 \tag{9.3–66}$$

In other words, $\delta^2(\boldsymbol{\varepsilon})$ is expressed as a linear combination of the squared projections of the channel vector $\boldsymbol{f}$ onto the eigenvectors of A. Each squared projection of the sum is weighted by the corresponding eigenvalue $\mu_k(\boldsymbol{\varepsilon})$, $k = 1, 2, \ldots, L+1$. Then

$$\delta^2_{\min} = \min_{\boldsymbol{\varepsilon}} \delta^2(\boldsymbol{\varepsilon}) \tag{9.3–67}$$

It is interesting to note that the worst channel characteristic of a given length $L+1$ can be obtained by finding the eigenvector corresponding to the minimum eigenvalue. Thus, if $\mu_{\min}(\boldsymbol{\varepsilon})$ is the minimum eigenvalue for a given error event $\boldsymbol{\varepsilon}$ and $\boldsymbol{v}_{\min}(\boldsymbol{\varepsilon})$ is the corresponding eigenvector, then

$$\begin{aligned}\mu_{\min} &= \min_{\boldsymbol{\varepsilon}} \mu_{\min}(\boldsymbol{\varepsilon})\\ \boldsymbol{f} &= \min_{\boldsymbol{\varepsilon}} \boldsymbol{v}_{\min}(\boldsymbol{\varepsilon})\end{aligned}$$

and

$$\delta^2_{\min} = \mu_{\min}$$

EXAMPLE 9.3-5. Let us determine the worst time-dispersive channel of length 3 ($L = 2$) by finding the minimum eigenvalue of A for different error events. Thus,

$$F(z) = f_0 + f_1 z^{-1} + f_2 z^{-2}$$

where f_0, f_1, and f_2 are the components of the eigenvector of A corresponding to the minimum eigenvalue. An error event of the form

$$\varepsilon(z) = 1 - z^{-1}$$

results in a matrix

$$A = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

which has the eigenvalues $\mu_1 = 2$, $\mu_2 = 2 + \sqrt{2}$, $\mu_3 = 2 - \sqrt{2}$. The eigenvector corresponding to μ_3 is

$$\boldsymbol{v}_3^t = \begin{bmatrix} \frac{1}{2} & \sqrt{\frac{1}{2}} & \frac{1}{2} \end{bmatrix} \tag{9.3–68}$$

We may also consider the dual error event

$$\boldsymbol{\varepsilon}(z) = 1 + z^{-1}$$

which results in the matrix

$$A = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

This matrix has eigenvalues identical to those of the one for $\varepsilon(z) = 1 - z^{-1}$. The corresponding eigenvector for $\mu_3 = 2 - \sqrt{2}$ is

$$\boldsymbol{v}_3^t = \begin{bmatrix} -\frac{1}{2} & \sqrt{\frac{1}{2}} & -\frac{1}{2} \end{bmatrix} \tag{9.3–69}$$

Any other error events lead to larger values for $\mu_{\min}$. Hence, $\mu_{\min} = 2 - \sqrt{2}$ and the worst-case channel is either

$$\begin{bmatrix} \frac{1}{2} & \sqrt{\frac{1}{2}} & \frac{1}{2} \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} -\frac{1}{2} & \sqrt{\frac{1}{2}} & -\frac{1}{2} \end{bmatrix}$$

The loss in SNR from the channel is

$$-10 \log \delta_{\min}^2 = -10 \log \mu_{\min} = 2.3\,\text{dB}$$

Repetitions of the above computation for channels with $L = 3, 4$, and 5 yield the results given in Table 9.3–1.

TABLE 9.3–1

Maximum Performance Loss and Corresponding Channel Characteristics

Channel length $L+1$	Performance loss $-10 \log \delta_{\min}^2$ dB	Minimum-distance channel
3	2.3	0.50, 0.71, 0.50
4	4.2	0.38, 0.60, 0.60, 0.38
5	5.7	0.29, 0.50, 0.58, 0.50, 0.29
6	7.0	0.23, 0.42, 0.52, 0.52, 0.42, 0.23

9.4
LINEAR EQUALIZATION

The MLSE for a channel with ISI has a computational complexity that grows exponentially with the length of the channel time dispersion. If the size of the symbol alphabet is M and the number of interfering symbols contributing to ISI is L, the Viterbi algorithm computes M^{L+1} metrics for each new received symbol. In most channels of practical interest, such a large computational complexity is prohibitively expensive to implement.

In this and the following sections, we describe suboptimum channel equalization approaches to compensate for the ISI. One approach employs a linear transversal filter, which is described in this section. This filter structure has a computational complexity that is a linear function of the channel dispersion length L.

The linear filter most often used for equalization is the transversal filter shown in Figure 9.4–1. Its input is the sequence $\{v_k\}$ given in Equation 9.3–16 and its output in the estimate of the information sequence $\{I_k\}$. The estimate of the kth symbol may be expressed as

$$\hat{I}_k = \sum_{j=-K}^{K} c_j v_{k-j} \tag{9.4–1}$$

where $\{c_j\}$ are the $2K + 1$ complex-valued tap weight coefficients of the filter. The estimate $\hat{I}_k$ is quantized to the nearest (in distance) information symbol to form the decision $\tilde{I}_k$. If $\tilde{I}_k$ is not identical to the transmitted information symbol I_k, an error has been made.

Considerable research has been performed on the criterion for optimizing the filter coefficients $\{c_k\}$. Since the most meaningful measure of performance for a digital communication system is the average probability of error, it is desirable to choose the coefficients to minimize this performance index. However, the probability of error is a highly non-linear function of $\{c_j\}$. Consequently, the probability of error as a performance

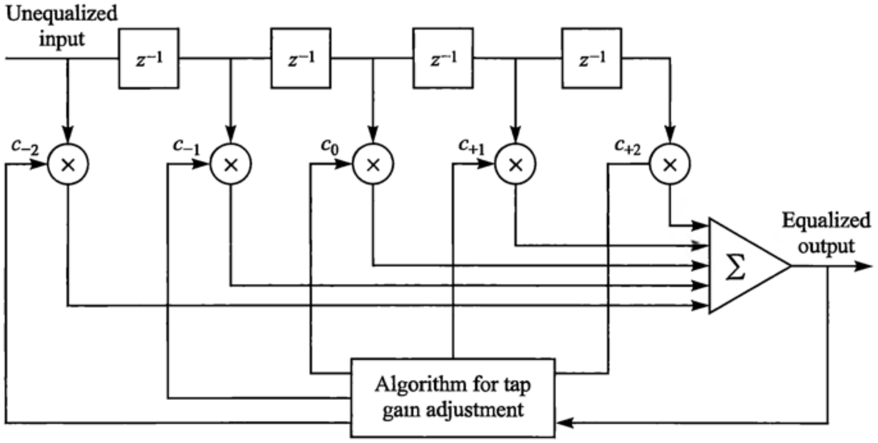


FIGURE 9.4–1
Linear transversal filter.

index for optimizing the tap weight coefficients of the equalizer is computationally complex.

Two criteria have found widespread use in optimizing the equalizer coefficients $\{c_j\}$. One is the peak distortion criterion and the other is the mean-square-error criterion.

9.4–1 Peak Distortion Criterion

The peak distortion is simply defined as the worst-case intersymbol interference at the output of the equalizer. The minimization of this performance index is called the *peak distortion criterion*. First we consider the minimization of the peak distortion assuming that the equalizer has an infinite number of taps. Then we shall discuss the case in which the transversal equalizer spans a finite time duration.

We observe that the cascade of the discrete-time linear filter model having an impulse response $\{f_n\}$ and an equalizer having an impulse response $\{c_n\}$ can be represented by a single equivalent filter having the impulse response

$$q_n = \sum_{j=-\infty}^{\infty} c_j f_{n-j} \tag{9.4–2}$$

That is, $\{q_n\}$ is simply the convolution of $\{c_n\}$ and $\{f_n\}$. The equalizer is assumed to have an infinite number of taps. Its output at the kth sampling instant can be expressed in the form

$$\hat{I}_k = q_0 I_k + \sum_{n\neq k} I_n q_{k-n} + \sum_{j=-\infty}^{\infty} c_j \eta_{k-j} \tag{9.4–3}$$

The first term in Equation 9.4–3 represents a scaled version of the desired symbol. For convenience, we normalize q_0 to unity. The second term is the intersymbol interference. The peak value of this interference, which is called the *peak distortion*, is

$$\begin{aligned} \mathcal{D}(\boldsymbol{c}) &= \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} |q_n| \\ &= \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \left| \sum_{j=-\infty}^{\infty} c_j f_{n-j} \right| \end{aligned} \tag{9.4–4}$$

Thus, $\mathcal{D}(\boldsymbol{c})$ is a function of the equalizer tap weights.

With an equalizer having an infinite number of taps, it is possible to select the tap weights so that $\mathcal{D}(\boldsymbol{c}) = 0$, i.e., $q_n = 0$ for all n except $n = 0$. That is, the intersymbol interference can be completely eliminated. The values of the tap weights for accomplishing this goal are determined from the condition

$$q_n = \sum_{j=-\infty}^{\infty} c_j f_{n-j} = \begin{cases} 1 & (n = 0) \\ 0 & (n \neq 0) \end{cases} \tag{9.4–5}$$

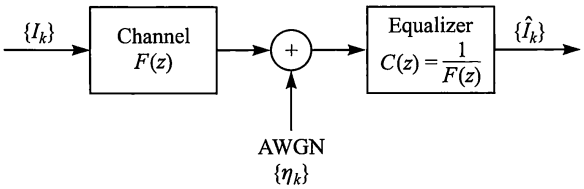

$\{I_k\}$
Channel
$F(z)$
+
Equalizer
$C(z) = \frac{1}{F(z)}$
$\{\hat{I}_k\}$
AWGN
$\{\eta_k\}$

FIGURE 9.4–2
Block diagram of channel with zero-forcing equalizer.

By taking the z transform of Equation 9.4–5, we obtain

$$Q(z) = C(z)F(z) = 1 \tag{9.4–6}$$

or, simply,

$$C(z) = \frac{1}{F(z)} \tag{9.4–7}$$

where $C(z)$ denotes the z transform of the $\{c_j\}$. Note that the equalizer, with transfer function $C(z)$, is simply the inverse filter to the linear filter model $F(z)$. In other words, complete elimination of the intersymbol interference requires the use of an inverse filter to $F(z)$. We call such a filter a *zero-forcing filter*. Figure 9.4–2 illustrates in block diagram the equivalent discrete-time channel and equalizer.

The cascade of the noise-whitening filter having the transfer function $1/F^*(1/z^*)$ and the zero-forcing equalizer having the transfer function $1/F(z)$ results in an equivalent zero-forcing equalizer having the transfer function

$$C'(z) = \frac{1}{F(z)F^*(1/z^*)} = \frac{1}{X(z)} \tag{9.4–8}$$

as shown in Figure 9.4–3. This combined filter has as its input the sequence $\{y_k\}$ of samples from the matched filter, given by Equation 9.3–10. Its output consists of the desired symbols corrupted only by additive zero-mean Gaussian noise. The impulse response of the combined filter is

$$\begin{aligned} c'_k &= \frac{1}{2\pi j}\oint C'(z)z^{k-1}dz \\ &= \frac{1}{2\pi j}\oint \frac{z^{k-1}}{X(z)}dz \end{aligned} \tag{9.4–9}$$

where the integration is performed on a closed contour that lies within the region of convergence of $C'(z)$. Since $X(z)$ is a polynomial with $2L$ roots $(\rho_1, \rho_2, \ldots, \rho_L, 1/\rho_1^*, 1/\rho_2^*, \ldots, 1/\rho_L^*)$, it follows that $C'(z)$ must converge in an annular region in the z plane

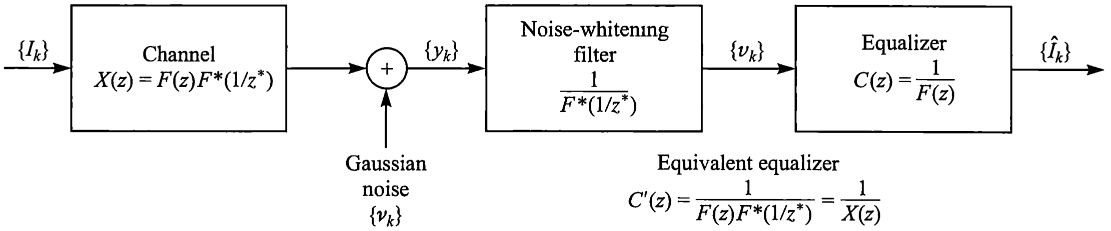

$\{I_k\}$
Channel
$X(z) = F(z)F^*(1/z^*)$
+
$\{y_k\}$
Noise-whitening filter
$\frac{1}{F^*(1/z^*)}$
$\{v_k\}$
Equalizer
$C(z) = \frac{1}{F(z)}$
$\{\hat{I}_k\}$
Gaussian noise
$\{\nu_k\}$
Equivalent equalizer
$C'(z) = \frac{1}{F(z)F^*(1/z^*)} = \frac{1}{X(z)}$

FIGURE 9.4–3
Block diagram of channel with equivalent zero-forcing equalizer.

that includes the unit circle ($z = e^{j\theta}$). Consequently, the closed contour in the integral can be the unit circle.

The performance of the infinite-tap equalizer that completely eliminates the intersymbol interference can be expressed in terms of the SNR at its output. For mathematical convenience, we normalize the received signal energy to unity.† This implies that $q_0 = 1$ and that the expected value of $|I_k|^2$ is also unity. Then the SNR is simply the reciprocal of the noise variance σ_n^2 at the output of the equalizer.‡

The value of σ_n^2 can be simply determined by observing that the noise sequence $\{\nu_k\}$ at the input to the equivalent zero-forcing equalizer $C'(z)$ has zero-mean and a power spectral density

$$\mathcal{S}_{\nu\nu}(\omega) = N_0 X(e^{j\omega T}), \qquad |\omega| \le \frac{\pi}{T} \tag{9.4–10}$$

where $X(e^{j\omega t})$ is obtained from $X(z)$ by the substitution $z = e^{j\omega T}$. Since $C'(z) = 1/X(z)$, it follows that the noise sequence at the output of the equalizer has a power spectral density

$$\mathcal{S}_{nn}(\omega) = \frac{N_0}{X(e^{j\omega T})}, \qquad |\omega| \le \frac{\pi}{T} \tag{9.4–11}$$

Consequently, the variance of the noise variable at the output of the equalizer is

$$\begin{aligned} \sigma_n^2 &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \mathcal{S}_{nn}(\omega)\, d\omega \\ &= \frac{T N_0}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{d\omega}{X(e^{j\omega T})} \end{aligned} \tag{9.4–12}$$

and the SNR for the zero-forcing equalizer is

$$\gamma_\infty = \frac{1}{\sigma_n^2} = \left[\frac{T N_0}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{d\omega}{X(e^{j\omega T})} \right]^{-1} \tag{9.4–13}$$

where the subscript on γ indicates that the equalizer has an infinite number of taps.

The spectral characteristics $X(e^{j\omega T})$ corresponding to the Fourier transform of the sampled sequence $\{x_n\}$ has an interesting relationship to the analog filter $H(\omega)$ used at the receiver. Since

$$x_k \doteq \int_{-\infty}^{\infty} h^*(t) h(t + kT)\, dt$$

use of Parseval's theorem yields

$$x_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} |H(\omega)|^2 e^{j\omega k T}\, d\omega \tag{9.4–14}$$

---

†This normalization is used throughout this chapter for mathematical convenience.

‡If desired, one can multiply this normalized SNR at the output of the equalizer by the signal energy.

where $H(\omega)$ is the Fourier transform of $h(t)$. But the integral in Equation 9.4–14 can be expressed in the form

$$x_k = \frac{1}{2\pi}\int_{-\pi/T}^{\pi/T}\left[\sum_{n=-\infty}^{\infty}\left|H\left(\omega+\frac{2\pi n}{T}\right)\right|^2\right]e^{j\omega kT}d\omega \tag{9.4–15}$$

Now, the Fourier transform of $\{x_k\}$ is

$$X(e^{j\omega T}) = \sum_{k=-\infty}^{\infty} x_k e^{-j\omega kT} \tag{9.4–16}$$

and the inverse transform yields

$$x_k = \frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} X(e^{j\omega T})e^{j\omega kT}d\omega \tag{9.4–17}$$

From a comparison of Equations 9.4–15 and 9.4–17, we obtain the desired relationship beween $X(e^{j\omega T})$ and $H(\omega)$. That is,

$$X(e^{j\omega T}) = \frac{1}{T}\sum_{n=-\infty}^{\infty}\left|H\left(\omega+\frac{2\pi n}{T}\right)\right|^2, \qquad |\omega| \le \frac{\pi}{T} \tag{9.4–18}$$

where the right-hand side of Equation 9.4–18 is called the*folded spectrum* of $|H(\omega)|^2$. We also observe that $|H(\omega)|^2 = X(\omega)$, where $X(\omega)$ is the Fourier transform of the waveform $x(t)$ and $x(t)$ is the response of the matched filter to the input pulse $h(t)$. Therefore the right-hand side of Equation 9.4–18 can also be expressed in terms of $X(\omega)$.

Substitution for $X(e^{j\omega T})$ in Equation 9.4–13 using the result in Equation 9.4–18 yields the desired expression for the SNR in the form

$$\gamma_\infty = \left[\frac{T^2 N_0}{2\pi}\int_{-\pi/T}^{\pi/T}\frac{d\omega}{\sum_{n=-\infty}^{\infty}|H(\omega+2\pi n/T)|^2}\right]^{-1} \tag{9.4–19}$$

We observe that if the folded spectral characteristic of $H(\omega)$ possesses any zeros, the integrand becomes infinite and the SNR goes to zero. In other words, the performance of the equalizer is poor whenever the folded spectral characteristic possesses nulls or takes on small values. This behavior occurs primarily because the equalizer, in eliminating the intersymbol interference, enhances the additive noise. For example, if the channel contains a spectral null in its frequency response, the linear zero-forcing equalizer attempts to compensate for this by introducing an infinite gain at that frequency. But this compensates for the channel distortion at the expense of enhancing the additive noise. On the other hand, an ideal channel coupled with an appropriate signal design that results in no intersymbol interference will have a folded spectrum that satisfies the condition

$$\sum_{n=-\infty}^{\infty}\left|H\left(\omega+\frac{2\pi n}{T}\right)\right|^2 = T, \qquad |\omega| \le \frac{\pi}{T} \tag{9.4–20}$$

In this case, the SNR achieves its maximum value, namely,

$$\gamma_\infty = \frac{1}{N_0} \tag{9.4–21}$$

Finite-length equalizer Let us now turn our attention to an equalizer having $2K+1$ taps. Since $c_j = 0$ for $|j| > K$, the convolution of $\{f_n\}$ with $\{c_n\}$ is zero outside the range $-K \le n \le K + L - 1$. That is, $q_n = 0$ for $n < -K$ and $n > K + L - 1$. With q_0 normalized to unity, the peak distortion is

$$\mathcal{D}(\boldsymbol{c}) = \sum_{\substack{n=-K \\ n\neq 0}}^{K+L-1} |q_n| = \sum_{\substack{n=-K \\ n\neq 0}}^{K+L-1} \left| \sum_j c_j f_{n-j} \right| \tag{9.4–22}$$

Although the equalizer has $2K + 1$ adjustable parameters, there are $2K + L$ nonzero values in the response $\{q_n\}$. Therefore, it is generally impossible to completely eliminate the intersymbol interference at the output of the equalizer. There is always some residual interference when the optimum coefficients are used. The problem is to minimize $\mathcal{D}(\mathbf{c})$ with respect to the coefficients $\{c_j\}$.

The peak distortion given by Equation 9.4–22 has been shown by Lucky (1965) to be a convex function of the coefficients $\{c_j\}$. That is, it possesses a global minimum and no local minima. Its minimization can be carried out numerically using, for example, the method of steepest descent. Little more can be said for the general solution to this minimization problem. However, for one special but important case, the solution for the minimization of $\mathcal{D}(\boldsymbol{c})$ is known. This is the case in which the distortion at the input to the equalizer, defined as

$$D_0 = \frac{1}{|f_0|} \sum_{n=1}^{L} |f_n| \tag{9.4–23}$$

is less than unity. This condition is equivalent to having the eye open prior to equalization. That is, the intersymbol interference is not severe enough to close the eye. Under this condition, the peak distortion $\mathcal{D}(\boldsymbol{c})$ is minimized by selecting the equalizer coefficients to force $q_n = 0$ for $1 \le |n| \le K$ and $q_0 = 1$. In other words, the general solution to the minimization of $\mathcal{D}(\boldsymbol{c})$, when $D_0 < 1$, is the zero-forcing solution for $\{q_n\}$ in the range $1 \le |n| \le K$. However, the values of $\{q_n\}$ for $K+1 \le n \le K+L-1$ are nonzero, in general. These nonzero values constitute the residual intersymbol interference at the output of the equalizer.

9.4–2 Mean-Square-Error (MSE) Criterion

In the MSE criterion, the tap weight coefficients $\{c_j\}$ of the equalizer are adjusted to minimize the mean square value of the error

$$\varepsilon_k = I_k - \hat{I}_k \tag{9.4–24}$$

where I_k is the information symbol transmitted in the kth signaling interval and $\hat{I}_k$ is the estimate of that symbol at the output of the equalizer, defined in Equation 9.4–1. When the information symbols $\{I_k\}$ are complex-valued, the performance index for the MSE criterion, denoted by J, is defined as

$$J = E|\varepsilon_k|^2 = E|I_k - \hat{I}_k|^2 \tag{9.4–25}$$

On the other hand, when the information symbols are real-valued, the performance index is simply the square of the real part of ε_k. In either case, J is a quadratic function of the equalizer coefficients $\{c_j\}$. In the following discussion, we consider the minimization of the complex-valued form given in Equation 9.4–25.

Infinite-length equalizer First, we shall derive the tap weight coefficients that minimize J when the equalizer has an infinite number of taps. In this case, the estimate $\hat{I}_k$ is expressed as

$$\hat{I}_k = \sum_{j=-\infty}^{\infty} c_j v_{k-j} \tag{9.4–26}$$

Substitution of Equation 9.4–26 into the expression for J given in Equation 9.4–25 and expansion of the result yields a quadratic function of the coefficients $\{c_j\}$. This function can be easily minimized with respect to the $\{c_j\}$ to yield a set (infinite in number) of linear equations for the $\{c_j\}$. Alternatively, the set of linear equations can be obtained by invoking the orthogonality principle in mean square estimation. That is, we select the coefficients $\{c_j\}$ to render the error ε_k orthogonal to the signal sequence $\{v^*_{k-l}\}$ for $-\infty < l < \infty$. Thus,

$$E\left(\varepsilon_k v^*_{k-l}\right) = 0, \qquad -\infty < l < \infty \tag{9.4–27}$$

Substitution for ε_k in Equation 9.4–27 yields

$$E\left[\left(I_k - \sum_{j=-\infty}^{\infty} c_j v_{k-j}\right) v^*_{k-l}\right] = 0$$

or, equivalently,

$$\sum_{j=-\infty}^{\infty} c_j E\left(v_{k-j} v^*_{k-l}\right) = E\left(I_k v^*_{k-l}\right), \qquad -\infty < l < \infty \tag{9.4–28}$$

To evaluate the moments in Equation 9.4–28, we use the expression for v_k given in Equation 9.3–16. Thus, we obtain

$$\begin{aligned} E\left(v_{k-j} v^*_{k-l}\right) &= \sum_{n=0}^{L} f_n^* f_{n+l-j} + N_0 \delta_{lj} \\ &= \begin{cases} x_{l-j} + N_0 \delta_{lj} & (|l-j| \le L) \\ 0 & (\text{otherwise}) \end{cases} \end{aligned} \tag{9.4–29}$$

and

$$E\left(I_k v_{k-l}^*\right) = \begin{cases} f_{-l}^* & (-L \leq l \leq 0) \\ 0 & \text{(otherwise)} \end{cases} \tag{9.4–30}$$

Now, if we substitute Equations 9.4–29 and 9.4–30 into Equation 9.4–28 and take the z transform of both sides of the resulting equation, we obtain

$$C(z)[F(z)F^*(1/z^*) + N_0] = F^*(1/z^*) \tag{9.4–31}$$

Therefore, the transfer function of the equalizer based on the MSE criterion is

$$C(z) = \frac{F^*(1/z^*)}{F(z)F^*(1/z^*) + N_0} \tag{9.4–32}$$

When the noise-whitening filter is incorporated into $C(z)$, we obtain an equivalent equalizer having the transfer function

$$\begin{aligned} C'(z) &= \frac{1}{F(z)F^*(1/z^*) + N_0} \\ &= \frac{1}{X(z) + N_0} \end{aligned} \tag{9.4–33}$$

We observe that the only difference between this expression for $C'(z)$ and the one based on the peak distortion criterion is the noise spectral density factor N_0 that appears in Equation 9.4–33. When N_0 is very small in comparison with the signal, the coefficients that minimize the peak distortion $\mathcal{D}(\boldsymbol{c})$ are approximately equal to the coefficients that minimize the MSE performance index J. That is, in the limit as $N_0 \rightarrow 0$, the two criteria yield the same solution for the tap weights. Consequently, when $N_0 = 0$, the minimization of the MSE results in complete elimination of the intersymbol interference. On the other hand, that is not the case when $N_0 \neq 0$. In general, when $N_0 \neq 0$, there is both residual intersymbol interference and additive noise at the output of the equalizer.

A measure of the residual intersymbol interference and additive noise is obtained by evaluating the minimum value of J, denoted by $J_{\min}$, when the transfer function $C(z)$ of the equalizer is given by Equation 9.4–32. Since $J = E|\varepsilon_k|^2 = E\left(\varepsilon_k I_k^*\right) - E\left(\varepsilon_k \hat{I}_k^*\right)$, and since $E\left(\varepsilon_k \hat{I}_k^*\right) = 0$ by virtue of the orthogonality conditions given in Equation 9.4–27, it follows that

$$\begin{aligned} J_{\min} &= E\left(\varepsilon_k I_k^*\right) \\ &= E|I_k|^2 - \sum_{j=-\infty}^{\infty} c_j E\left(v_{k-j} I_k^*\right) \\ &= 1 - \sum_{j=-\infty}^{\infty} c_j f_{-j} \end{aligned} \tag{9.4–34}$$

This particular form for $J_{\min}$ is not very informative. More insight on the performance of the equalizer as a function of the channel characteristics is obtained when the summation in Equation 9.4–34 is transformed into the frequency domain. This can be accomplished by first noting that the summation in Equation 9.4–34 is the convolution of $\{c_j\}$ with $\{f_j\}$, evaluated at a shift of zero. Thus, if $\{b_k\}$ denotes the convolution of these two sequences, the summation in Equation 9.4–34 is simply equal to b_0. Since the z transform of the sequence $\{b_k\}$ is

$$\begin{aligned} B(z) &= C(z)F(z) \\ &= \frac{F(z)F^*(1/z^*)}{F(z)F^*(1/z^*) + N_0} \\ &= \frac{X(z)}{X(z) + N_0} \end{aligned} \tag{9.4–35}$$

the term b_0 is

$$\begin{aligned} b_0 &= \frac{1}{2\pi j} \oint \frac{B(z)}{z} dz \\ &= \frac{1}{2\pi j} \oint \frac{X(z)}{z[X(z) + N_0]} dz \end{aligned} \tag{9.4–36}$$

The contour integral in Equation 9.4–36 can be transformed into an equivalent line integral by the change of variable $z = e^{j\omega T}$. The result of this change of variable is

$$b_0 = \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{X(e^{j\omega T})}{X(e^{j\omega T}) + N_0} d\omega \tag{9.4–37}$$

Finally, substitution of the result in Equation 9.4–37 for the summation in Equation 9.4–34 yields the desired expression for the minimum MSE in the form

$$\begin{aligned} J_{\min} &= 1 - \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{X(e^{j\omega T})}{X(e^{j\omega T}) + N_0} d\omega \\ &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{N_0}{X(e^{j\omega T}) + N_0} d\omega \\ &= \frac{T}{2\pi} \int_{-\pi/T}^{\pi/T} \frac{N_0}{T^{-1} \sum_{n=-\infty}^{\infty} |H(\omega + 2\pi n/T)|^2 + N_0} d\omega \end{aligned} \tag{9.4–38}$$

In the absence of intersymbol interference, $X(e^{j\omega T}) = 1$ and, hence,

$$J_{\min} = \frac{N_0}{1 + N_0} \tag{9.4–39}$$

We observe that $0 \leq J_{\min} \leq 1$. Furthermore, the relationship between the output (normalized by the signal energy) SNR γ_∞ and $J_{\min}$ must be

$$\gamma_\infty = \frac{1 - J_{\min}}{J_{\min}} \tag{9.4–40}$$

More importantly, this relation between γ_∞ and $J_{\min}$ also holds when there is residual intersymbol interference in addition to the noise.

Finite-length equalizer Let us now turn our attention to the case in which the transversal equalizer spans a finite time duration. The output of the equalizer in the kth signaling interval is

$$\hat{I}_k = \sum_{j=-K}^{K} c_j v_{k-j} \tag{9.4–41}$$

The MSE for the equalizer having $2K + 1$ taps, denoted by $J(K)$, is

$$J(K) = E|I_k - \hat{I}_k|^2 = E\left|I_k - \sum_{j=-K}^{K} c_j v_{k-j}\right|^2 \tag{9.4–42}$$

Minimization of $J(K)$ with respect to the tap weights $\{c_j\}$ or, equivalently, forcing the error $\varepsilon_k = I_k - \hat{I}_k$ to be orthogonal to the signal samples v^*_{j-l}, $|l| \le K$, yields the following set of simultaneous equations:

$$\sum_{j=-K}^{K} c_j \Gamma_{lj} = \xi_l, \qquad l = -K, \ldots, -1, 0, 1, \ldots, K \tag{9.4–43}$$

where

$$\Gamma_{lj} = \begin{cases} x_{l-j} + N_0 \delta_{lj} & (|l - j\,| \le L) \\ 0 & \text{(otherwise)} \end{cases} \tag{9.4–44}$$

and

$$\xi_l = \begin{cases} f^*_{-l} & (-L \le l \le 0) \\ 0 & \text{(otherwise)} \end{cases} \tag{9.4–45}$$

It is convenient to express the set of linear equations in matrix form. Thus,

$$\boldsymbol{\Gamma C} = \boldsymbol{\xi} \tag{9.4–46}$$

where $\boldsymbol{C}$ denotes the column vector of $2K + 1$ tap weight coefficients, $\boldsymbol{\Gamma}$ denotes the $(2K + 1) \times (2K + 1)$ Hermitian covariance matrix with elements Γ_{ij} and ξ is a $(2K + 1)$-dimensional column vector with elements ξ_i. The solution of Equation 9.4–46 is

$$\boldsymbol{C}_{\text{opt}} = \boldsymbol{\Gamma}^{-1}\boldsymbol{\xi} \tag{9.4–47}$$

Thus, the solution for $\boldsymbol{C}_{\text{opt}}$ involves inverting the matrix $\boldsymbol{\Gamma}$. The optimum tap weight coefficients given by Equation 9.4–47 minimize the performance index $J(K)$, with the result that the minimum value of $J(K)$ is

$$\begin{aligned} J_{\min}(K) &= 1 - \sum_{j=-K}^{0} c_j f_{-j} \\ &= 1 - \boldsymbol{\xi}^H \boldsymbol{\Gamma}^{-1} \boldsymbol{\xi} \end{aligned} \tag{9.4–48}$$

where H represents the conjugate transpose. $J_{\min}(K)$ may be used in Equation 9.4–40 to compute the output SNR for the linear equalizer with $2K + 1$ tap coefficients.

9.4–3 Performance Characteristics of the MSE Equalizer

In this section, we consider the performance characteristics of the linear equalizer that is optimized by using the MSE criterion. Both the minimum MSE and the probability of error are considered as performance measures for some specific channels. We begin by evaluating the minimum MSE $J_{\min}$ and the output SNR γ_∞ for two specific channels. Then, we consider the evaluation of the probability of error.

EXAMPLE 9.4–1. First, we consider an equivalent discrete-time channel model consisting of two components f_0 and f_1, which are normalized to $|f_0|^2 + |f_1|^2 = 1$. Then

$$F(z) = f_0 + f_1 z^{-1} \tag{9.4–49}$$

and

$$X(z) = f_0 f_1^* z + 1 + f_0^* f_1 z^{-1} \tag{9.4–50}$$

The corresponding frequency response is

$$\begin{aligned} X(e^{j\omega T}) &= f_0 f_1^* e^{j\omega T} + 1 + f_0^* f_1 e^{-j\omega T} \\ &= 1 + 2|f_0||f_1|\cos(\omega T + \theta) \end{aligned} \tag{9.4–51}$$

where θ is the angle of $f_0 f_1^*$. We note that this channel characteristic possesses a null at $\omega = \pi/T$ when $f_0 = f_1 = \sqrt{\frac{1}{2}}$.

A linear equalizer with an infinite number of taps, adjusted on the basis of the MSE criterion, will have the minimum MSE given by Equation 9.4–38. Evaluation of the integral in Equation 9.4–38 for the $X(e^{j\omega T})$ given in Equation 9.4–51 yields the result

$$\begin{aligned} J_{\min} &= \frac{N_0}{\sqrt{N_0^2 + 2N_0(|f_0|^2 + |f_1|^2) + (|f_0|^2 - |f_1|^2)^2}} \\ &= \frac{N_0}{\sqrt{N_0^2 + 2N_0 + (|f_0|^2 - |f_1|^2)^2}} \end{aligned} \tag{9.4–52}$$

Let us consider the special case in which $f_0 = f_1 = \sqrt{\frac{1}{2}}$. The minimum MSE is $J_{\min} = N_0/\sqrt{N_0^2 + 2N_0}$ and the corresponding output SNR is

$$\begin{aligned} \gamma_\infty &= \sqrt{1 + \frac{2}{N_0}} - 1 \\ &\approx \left(\frac{2}{N_0}\right)^{1/2}, \qquad N_0 \ll 1 \end{aligned} \tag{9.4–53}$$

This result should be compared with the output SNR of $1/N_0$ obtained in the case of no intersymbol interference. A significant loss in SNR occurs from this channel.

EXAMPLE 9.4–2. As a second example, we consider an exponentially decaying characteristic of the form

$$f_k = \sqrt{1 - a^2}\, a^k, \qquad k = 0, 1, \ldots$$

where $a < 1$. The Fourier transform of this sequence is

$$X(e^{j\omega T}) = \frac{1-a^2}{1+a^2-2a\cos\omega T} \tag{9.4–54}$$

which is a function that contains a minimum at $\omega = \pi/T$.

The output SNR for this channel is

$$\begin{aligned} \gamma_\infty &= \left(\sqrt{1+2N_0\frac{1+a^2}{1-a^2}+N_0^2}-1\right)^{-1} \\ &\approx \frac{1-a^2}{(1+a^2)N_0}, \qquad N_0 \ll 1 \end{aligned} \tag{9.4–55}$$

Therefore, the loss in SNR due to the presence of the interference is

$$-10\log_{10}\left(\frac{1-a^2}{1+a^2}\right)$$

Probability of error performance of linear MSE equalizer Above, we discussed the performance of the linear equalizer in terms of the minimum achievable MSE $J_{\min}$ and the output SNR γ that is related to $J_{\min}$ through the formula in Equation 9.4–40. Unfortunately, there is no simple relationship between these quantities and the probability of error. The reason is that the linear MSE equalizer contains some residual intersymbol interference at its output. This situation is unlike that of the infinitely long zero-forcing equalizer, for which there is no residual interference, but only Gaussian noise. The residual interference at the output of the MSE equalizer is not well characterized as an additional Gaussian noise term, and, hence, the output SNR does not translate easily into an equivalent error probability.

One approach to computing the error probability is a brute force method that yields an exact result. To illustrate this method, let us consider a PAM signal in which the information symbols are selected from the set of values $2n - M - 1$, $n = 1, 2, \ldots, M$, with equal probability. Now consider the decision on the symbol I_n. The estimate of I_n is

$$\hat{I}_n = q_0 I_n + \sum_{k\neq n} I_k q_{n-k} + \sum_{j=-K}^{K} c_j \eta_{n-j} \tag{9.4–56}$$

where $\{q_n\}$ represent the convolution of the impulse response of the equalizer and equivalent channel, i.e.,

$$q_n = \sum_{k=-K}^{K} c_k f_{n-k} \tag{9.4–57}$$

and the input signal to the equalizer is

$$v_k = \sum_{j=0}^{L} f_j I_{k-j} + \eta_k \tag{9.4–58}$$

The first term in the right-hand side of Equation 9.4–56 is the desired symbol, the middle term is the intersymbol interference, and the last term is the Gaussian noise. The variance of the noise is

$$\sigma_n^2 = N_0 \sum_{j=-K}^{K} c_j^2 \tag{9.4–59}$$

For an equalizer with $2K + 1$ taps and a channel response that spans $L + 1$ symbols, the number of symbols involved in the intersymbol interference is $2K + L$.

Define

$$\mathcal{D} = \sum_{k \neq n} I_k q_{n-k} \tag{9.4–60}$$

For a particular sequence of $2K + L$ information symbols, say the sequence $\boldsymbol{I}_J$, the intersymbol interference term $\mathcal{D} \equiv D_J$ is fixed. The probability of error for a fixed D_J is

$$\begin{aligned} P_e(D_J) &= 2\frac{M-1}{M} P(N + D_J > q_0) \\ &= \frac{2(M-1)}{M} Q\left(\sqrt{\frac{(q_0 - D_J)^2}{\sigma_n^2}}\right) \end{aligned} \tag{9.4–61}$$

where N denotes the additive noise term. The average probability of error is obtained by averaging $P_e(D_J)$ over all possible sequences $\boldsymbol{I}_J$. That is,

$$\begin{aligned} P_e &= \sum_{\mathbf{I}_J} P_e(D_J) P(\boldsymbol{I}_J) \\ &= \frac{2(M-1)}{M} \sum_{\mathbf{I}_J} Q\left(\sqrt{\frac{(q_0 - D_J)^2}{\sigma_n^2}}\right) P(\boldsymbol{I}_J) \end{aligned} \tag{9.4–62}$$

When all the sequences are equally likely,

$$P(\boldsymbol{I}_J) = \frac{1}{M^{2K+L}} \tag{9.4–63}$$

The conditional error probability terms $P_e(D_J)$ are dominated by the sequence that yields the largest value of D_J. This occurs when $I_n = \pm(M-1)$ and the signs of the information symbols match the signs of the corresponding $\{q_n\}$. Then,

$$D_J^* = (M-1) \sum_{k \neq 0} |q_k|$$

and

$$P_e(D_J^*) = \frac{2(M-1)}{M} Q\left(\sqrt{\frac{q_0^2}{\sigma_n^2}\left(1 - \frac{M-1}{q_0} \sum_{k \neq 0} |q_k|\right)^2}\right) \tag{9.4–64}$$

Thus, an upper bound on the average probability of error for equally likely symbol sequences is

$$P_e \le P_e(D_J^*) \tag{9.4–65}$$

If the computation of the exact error probability in Equation 9.4–62 proves to be too cumbersome and too time consuming because of the large number of terms in the sum and if the upper bound is too loose, one can resort to one of a number of different approximate methods that have been devised, which are known to yield tight bounds on P_e. A discussion of these different approaches would take us too far afield. The interested reader is referred to the papers by Saltzberg (1968), Lugannani (1969), Ho and Yeh (1970), Shimbo and Celebiler (1971), Glave (1972), Yao (1972), and Yao and Tobin (1976).

As an illustration of the performance limitations of a linear equalizer in the presence of severe intersymbol interference, we show in Figure 9.4–4 the probability of error for binary (antipodal) signaling, as measured by Monte Carlo simulation, for the three discrete-time channel characteristics shown in Figure 9.4–5. For purposes of comparison, the performance obtained for a channel with no intersymbol interference is also illustrated in Figure 9.4–4. The equivalent discrete-time channel shown in Figure 9.4–5a is typical of the response of a good-quality telephone channel. In contrast, the equivalent discrete-time channel characteristics shown in Figure 9.4–5b and c result

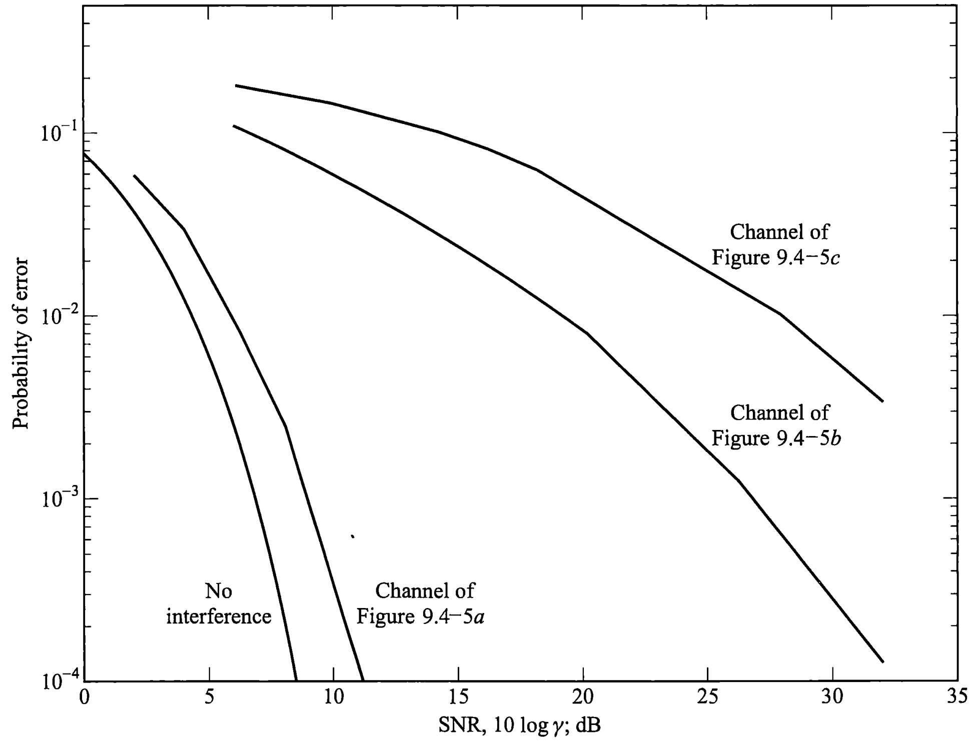


FIGURE 9.4–4
Error rate performance of linear MSE equalizer. Thirty-one taps in transversal equalizer. $\left(\gamma = \frac{1}{N_0}\sum_k |f_k|^2\right)$.

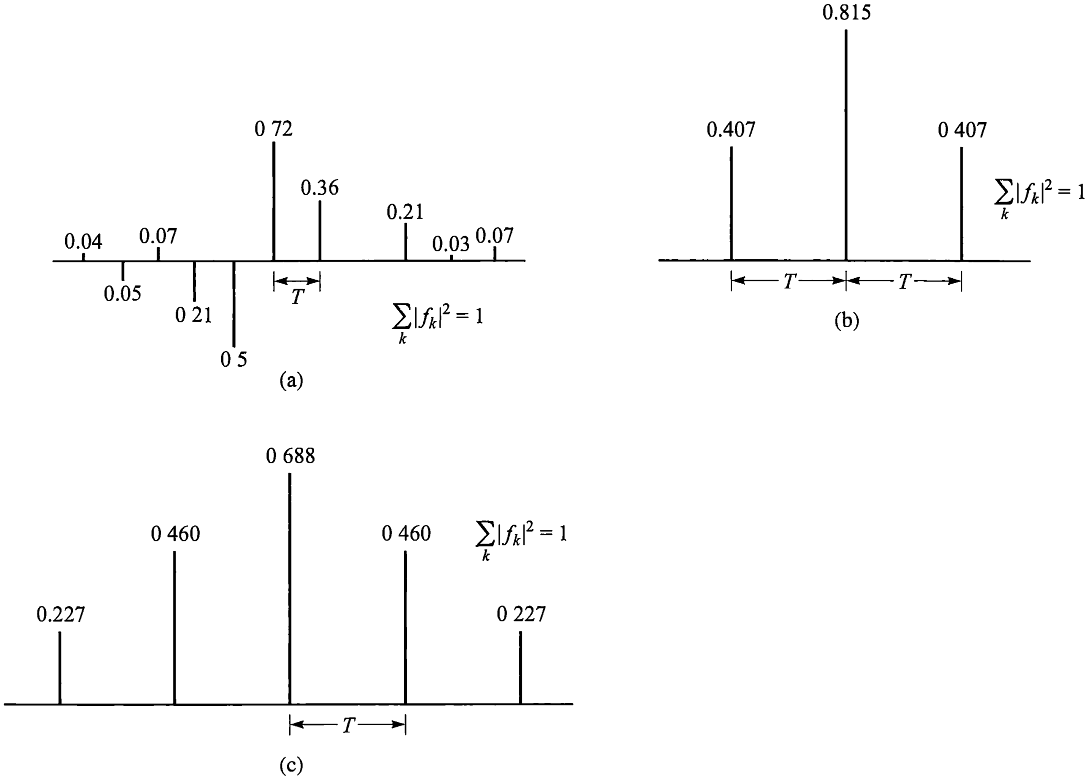
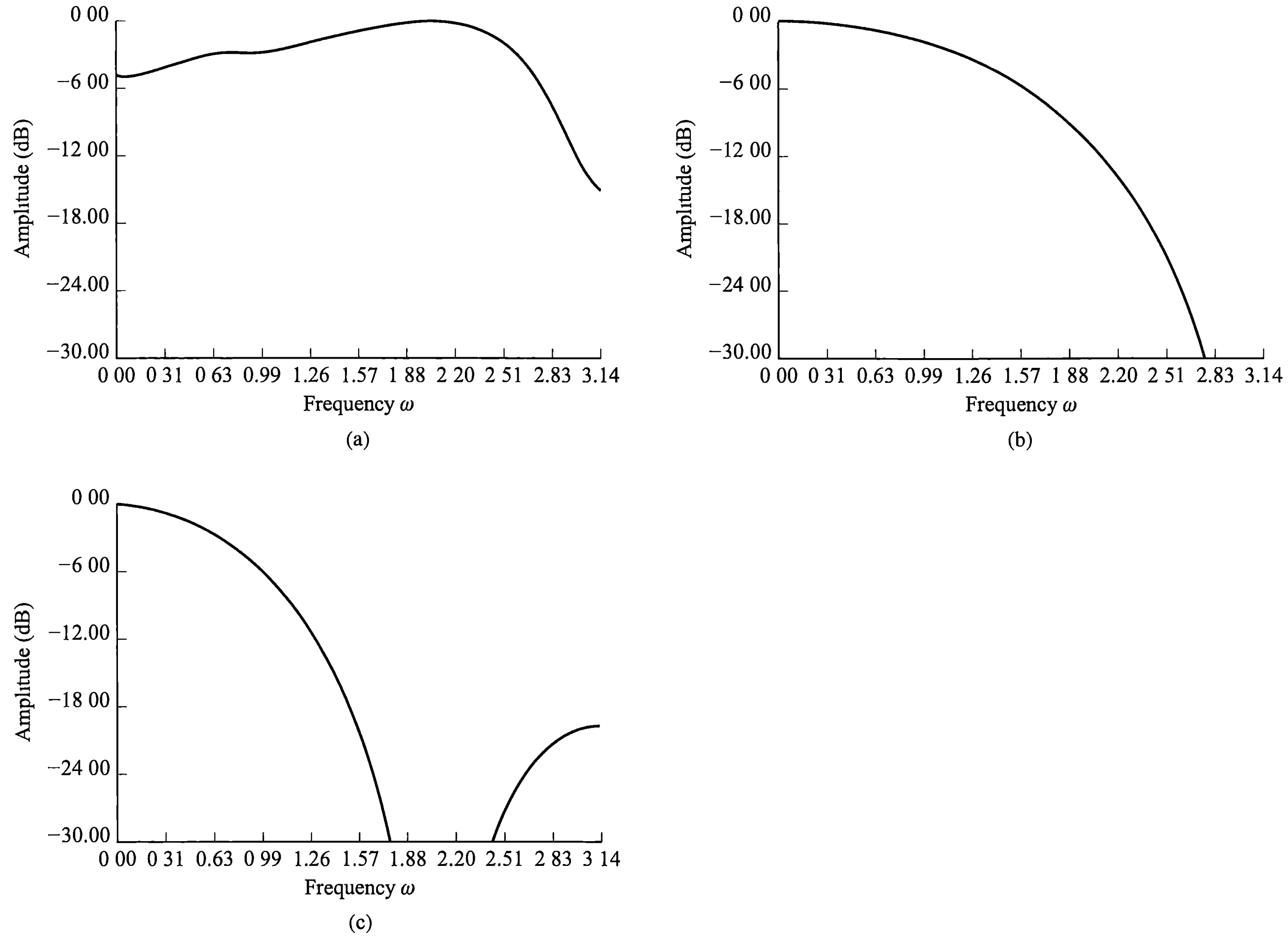


FIGURE 9.4–5
Three discrete-time channel characteristics.

in severe intersymbol interference. The spectral characteristics $|X(e^{j\omega})|$ for the three channels, illustrated in Figure 9.4–6, clearly show that the channel in Figure 9.4–5c has the worst spectral characteristic. Hence the performance of the linear equalizer for this channel is the poorest of the three cases. Next in performance is the channel shown in Figure 9.4–5b, and finally, the best performance is obtained with the channel shown in Fig. 9.4–5a. In fact, the error rate of the latter is within 3 dB of the error rate achieved with no interference.

One conclusion reached from the results on output SNR γ_∞ and the limited probability of error results illustrated in Figure 9.4–4 is that a linear equalizer yields good performance on channels such as telephone lines, where the spectral characteristics of the channels are well behaved and do not exhibit spectral nulls. On the other hand, a linear equalizer is inadequate as a compensator for the intersymbol interference on channels with spectral nulls, which may be encountered in radio transmission. In general, the channel spectral nulls result in a large noise enhancement at the output of the linear equalizer.

The basic limitation of the linear equalizer to cope with severe ISI has motivated a considerable amount of research into non-linear equalizers with low computational complexity. The decision-feedback equalizer described in Section 9.5 is shown to be an effective solution to this problem.



(a)

(b)

(c)

FIGURE 9.4–6
Amplitude spectra for the channels shown in Figure 9.4–5a, b, and c, respectively.

9.4–4 Fractionally Spaced Equalizers

In the linear equalizer structures that we have described in the previous section, the equalizer taps are spaced at the reciprocal of the symbol rate, i.e., at the reciprocal of the signaling rate $1/T$. This tap spacing is optimum if the equalizer is preceded by a filter matched to the channel distorted transmitted pulse. When the channel characteristics are unknown, the receiver filter is sometimes matched to the transmitted signal pulse and the sampling time is optimized for this suboptimum filter. In general, this approach leads to an equalizer performance that is very sensitive to the choice of sampling time.

The limitations of the symbol rate equalizer are most easily evident in the frequency domain. From Equation 9.2–5, the spectrum of the signal at the input to the equalizer may be expressed as

$$Y_T(f) = \frac{1}{T}\sum_n X\left(f - \frac{n}{T}\right) e^{j2\pi(f-n/T)\tau_0} \tag{9.4–66}$$

where $Y_T(f)$ is the folded or aliased spectrum, where the folding frequency is $1/2T$. Note that the received signal spectrum is dependent on the choice of the sampling delay τ_0. The signal spectrum at the output of the equalizer is $C_T(f)Y_T(f)$, where

$$C_T(f) = \sum_{k=-K}^{K} c_k e^{-j2\pi f k T} \tag{9.4–67}$$

It is clear from these relationships that the symbol rate equalizer can only compensate for the frequency-response characteristics of the aliased received signal. It cannot compensate for the channel distortion inherent in $X(f)e^{j2\pi f\tau_0}$.

In contrast to the symbol rate equalizer, a *fractionally spaced equalizer* (FSE) is based on sampling the incoming signal at least as fast as the Nyquist rate. For example, if the transmitted signal consists of pulses having a raised cosine spectrum with a roll-off factor β, its spectrum extends to $F_{\max} = (1+\beta)/2T$. This signal can be sampled at the receiver at a rate

$$2F_{\max} = \frac{1+\beta}{T} \tag{9.4–68}$$

and then passed through an equalizer with tap spacing of $T/(1+\beta)$. For example, if $\beta = 1$, we would have a $\frac{1}{2}T$-spaced equalizer. If $\beta = 0.5$, we would have a $\frac{2}{3}T$-spaced equalizer, and so forth. In general, then, a digitally implemented fractionally spaced equalizer has tap spacing of MT/N where M and N are integers and $N > M$. Usually, a $\frac{1}{2}T$-spaced equalizer is used in many applications.

Since the frequency response of the FSE is

$$C_{T'}(f) = \sum_{k=-K}^{K} c_k e^{-j2\pi f k T'} \tag{9.4–69}$$

where $T' = MT/N$, it follows that $C_{T'}(f)$ can equalize the received signal spectrum beyond the Nyquist frequency $f = 1/2T$ to $f = (1+\beta)/2T = N/2MT$. The equalized spectrum is

$$\begin{aligned} C_{T'}(f)Y_{T'}(f) &= C_{T'}(f)\sum_n X\left(f - \frac{n}{T'}\right) e^{j2\pi(f-n/T')\tau_0} \\ &= C_{T'}(f)\sum_n X\left(f - \frac{nN}{MT}\right) e^{j2\pi(f-nN/MT)\tau_0} \end{aligned} \tag{9.4–70}$$

Since $X(f) = 0$ for $|f| > N/2MT$, Equation 9.4–70 may be expressed as

$$C_{T'}(f)Y_{T'}(f) = C_{T'}(f)X(f)e^{j2\pi f\tau_0}, \qquad |f| \le \frac{1}{2T'} \tag{9.4–71}$$

Thus, we observe that the FSE compensates for the channel distortion in the received signal before the aliasing effects due to symbol rate sampling. In other words, $C_{T'}(f)$ can compensate for an arbitrary timing phase.

The FSE output is sampled at the symbol rate $1/T$ and has the spectrum

$$\sum_k C_{T'}\left(f - \frac{k}{T}\right) X\left(f - \frac{k}{T}\right) e^{j2\pi(f-k/T)\tau_0} \tag{9.4–72}$$

In effect, the optimum FSE is equivalent to the optimum linear receiver consisting of the matched filter followed by a symbol rate equalizer.

Let us now consider the adjustment of the tap coefficients in the FSE. The input to the FSE may be expressed as

$$y\left(\frac{kMT}{N}\right) = \sum_n I_n x\left(\frac{kMT}{N} - nT\right) + v\left(\frac{kMT}{N}\right) \tag{9.4–73}$$

In each symbol interval, the FSE produces an output of the form

$$\hat{I}_k = \sum_{n=-K}^{K} c_n y\left(kT - \frac{nMT}{N}\right) \tag{9.4–74}$$

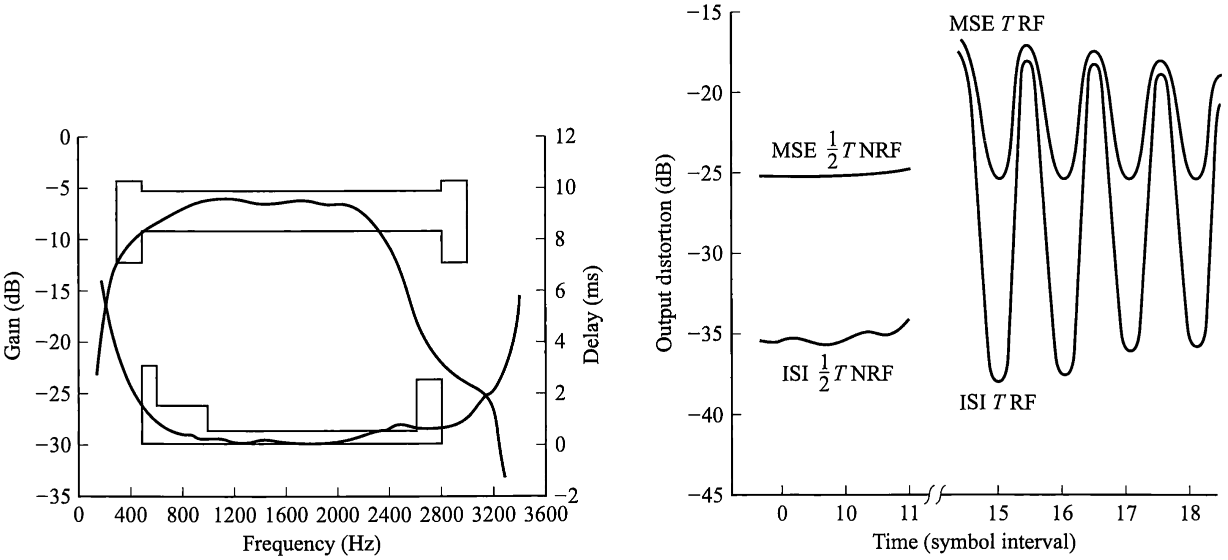
where the coefficients of the equalizer are selected to minimize the MSE. This optimization leads to a set of linear equations for the equalizer coefficients that have the solution

$$\boldsymbol{C}_{\text{opt}} = \boldsymbol{A}^{-1}\boldsymbol{\alpha} \tag{9.4–75}$$

where $\boldsymbol{A}$ is the covariance matrix of the input data and $\boldsymbol{\alpha}$ is the vector of cross correlations. These equations are identical in form to those for the symbol rate equalizer, but there are some subtle differences. One is that $\boldsymbol{A}$ is Hermitian, but not Toeplitz. In addition, $\boldsymbol{A}$ exhibits periodicities that are inherent in a cyclostationary process, as shown by Qureshi (1985). As a result of the fractional spacing, some of the eigenvalues of $\boldsymbol{A}$ are nearly zero. Attempts have been made by Long et al. (1988a, b) to exploit this property in the coefficient adjustment.

An analysis of the performance of fractionally spaced equalizers, including their convergence properties, is given in a paper by Ungerboeck (1976). Simulation results demonstrating the effectiveness of the FSE over a symbol rate equalizer have also been given in the papers by Qureshi and Forney (1977) and Gitlin and Weinstein (1981). We cite two examples from these papers. First, Figure 9.4–7 illustrates the performance of the symbol rate equalizer and a $\frac{1}{2}T$-FSE for a channel with high-end amplitude distortion, whose characteristics are also shown in this figure. The symbol-spaced equalizer was preceded with a filter matched to the transmitted pulse that had a (square-root) raised cosine spectrum with a 20 percent roll-off ($\beta = 0.2$). The FSE did not have any filter preceding it. The symbol rate was 2400 symbols/s and the modulation was QAM. The received SNR was 30 dB. Both equalizers had 31 taps; hence, the $\frac{1}{2}T$-FSE spanned one-half of the time interval of the symbol rate equalizer. Nevertheless, the FSE outperformed the symbol rate equalizer when the latter was optimized at the best sampling time. Furthermore, the FSE did not exhibit any sensitivity to timing phase, as illustrated in Figure 9.4–7b.

Similar results were obtained by Gitlin and Weinstein. For a channel with poor envelope delay characteristics, the SNR performance of the symbol rate equalizer and a $\frac{1}{2}T$-FSE are illustrated in Figure 9.4–8. In this case, both equalizers had the same time span. The T-spaced equalizer had 24 taps while the FSE had 48 taps. The symbol rate was 2400 symbols/s and the data rate was 9600 bits/s with 16-QAM modulation. The signal pulse had a raised cosine spectrum with $\beta = 0.12$. Note again that the FSE outperformed the T-spaced equalizer by several decibels, even when the latter was



(a) Channel with high-end amplitude distortion (HA)

(b) Equalizer performance

FIGURE 9.4–7

T and $\frac{1}{2}$ T equalizer performance as a function of timing phase for 2400 symbols per second. (NRF indicates no receiver filter.) [*From Qureshi and Forney (1977).* © *1977 IEEE.*]

adjusted for optimum sampling. The results in these two papers clearly demonstrate the superior performance achieved with a fractionally spaced equalizer.

9.4–5 Baseband and Passband Linear Equalizers

The linear equalizer treated above was described in terms of equivalent lowpass signals. However, in a practical implementation, the linear equalizer shown in Figure 9.4–1 can be realized either at baseband or at bandpass. For example, Figure 9.4–9 illustrates the demodulation of QAM or multiphase PSK by first translating the signal to baseband and equalizing the baseband signal with an equalizer having complex-valued coefficients. In effect, the equalizer with a complex-valued (in-phase and quadrature components) input

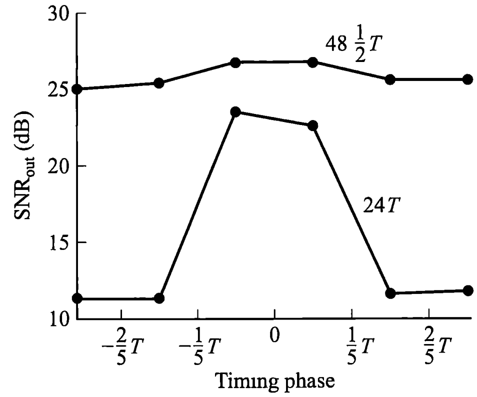
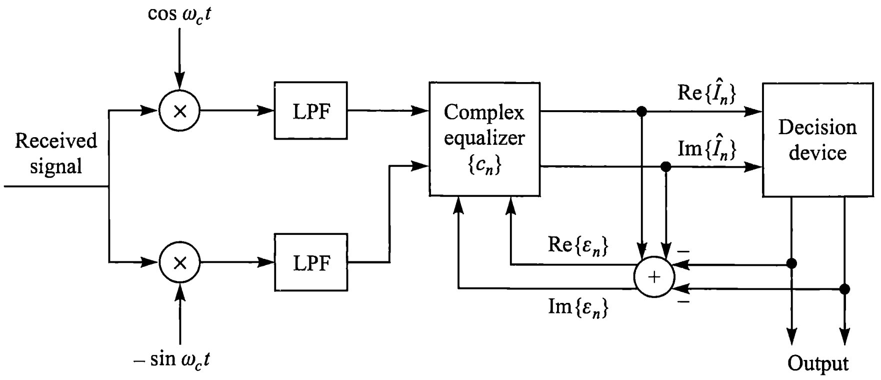


FIGURE 9.4–8

Performance of T and $\frac{1}{2}$ T equalizers as a function of timing phase for 2400 symbols/s 16-QAM on a channel with poor envelope delay. [*From Gitlin and Weinstein (1981). Reprinted with permission from* Bell System Technical Journal. © *1981 AT & T.*]



cos $\omega_c t$
Received signal
LPF
LPF
Complex equalizer $\{c_n\}$
$\mathrm{Re}\{\hat{I}_n\}$
$\mathrm{Im}\{\hat{I}_n\}$
Decision device
$\mathrm{Re}\{\varepsilon_n\}$
$\mathrm{Im}\{\varepsilon_n\}$
$-\sin \omega_c t$
Output

FIGURE 9.4–9
QAM and PSK signal demodulator with baseband equalizer.

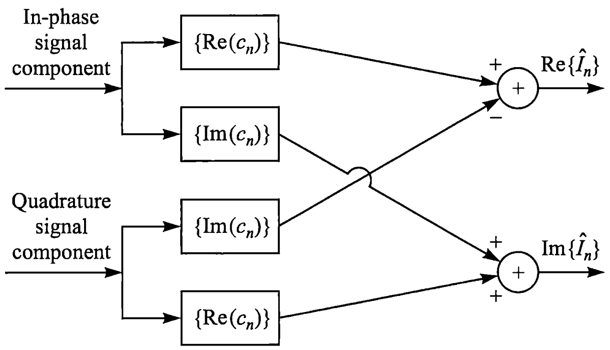
is equivalent to four parallel equalizers with real-valued tap coefficients as shown in Figure 9.4–10. We generally refer to the equalizer in Figure 9.4–9 as a complex-valued *baseband equalizer*.

As an alternative, we may equalize the signal at passband. This is accomplished as shown in Figure 9.4–11 for two-dimensional signal constellations such as QAM and PSK. The received signal is filtered and, in parallel, it is passed through a Hilbert transformer, called a *phase-splitting filter*. Thus, we have the equivalent of in-phase and quadrature components at passband, which are fed to a passband complex equalizer. We may call this equalizer structure a complex-valued *passband equalizer*. Following the equalization, the signal is down-converted to baseband and detected.

The complex-valued baseband equalizer may be implemented either as a symbol rate equalizer (SRE) or as a fractionally spaced equalizer (FSE), with the latter being preferable in view of its insensitivity to the sampling phase within a symbol interval.

The complex-valued passband equalizer must be an FSE, with samples of the received signal taken at some multiple of the symbol rate that exceeds the Nyquist rate.

An alternative passband FSE to the structure shown in Figure 9.4–11 is illustrated in Figure 9.4–12. In this FSE, real-valued samples of the received signal are taken at the Nyquist rate or faster and equalized at bandpass by a linear equalizer that has complex-valued coefficients. We note that this equalizer structure does not explicitly



In-phase signal component
$\{\mathrm{Re}(c_n)\}$
$\{\mathrm{Im}(c_n)\}$
Quadrature signal component
$\{\mathrm{Im}(c_n)\}$
$\{\mathrm{Re}(c_n)\}$
+
−
$\mathrm{Re}\{\hat{I}_n\}$
+
+
$\mathrm{Im}\{\hat{I}_n\}$

FIGURE 9.4–10
Complex-valued baseband equalizer for QAM and PSK signals.

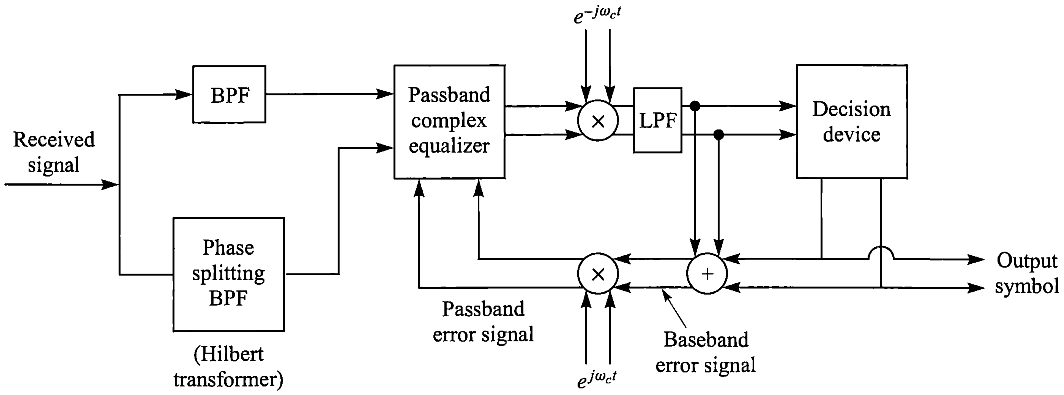

$e^{-j\omega_c t}$
Received signal
BPF
Phase splitting BPF
(Hilbert transformer)
Passband complex equalizer
×
LPF
Decision device
+
×
Output symbol
Passband error signal
$e^{j\omega_c t}$
Baseband error signal

FIGURE 9.4–11
QAM or PSK signal equalization at passband.

implement a Hilbert transformer to perform phase splitting. Instead, the phase-splitting function is embedded in the equalizer coefficients and, thus, the Hilbert transform is avoided. This alternative passband FSE structure in Figure 9.4–12 has been called a *phase-splitting FSE* (PS-FSE). Its properties and its performance has been investigated by Mueller and Werner (1982), Im and Un (1987), and Ling and Qureshi (1990).

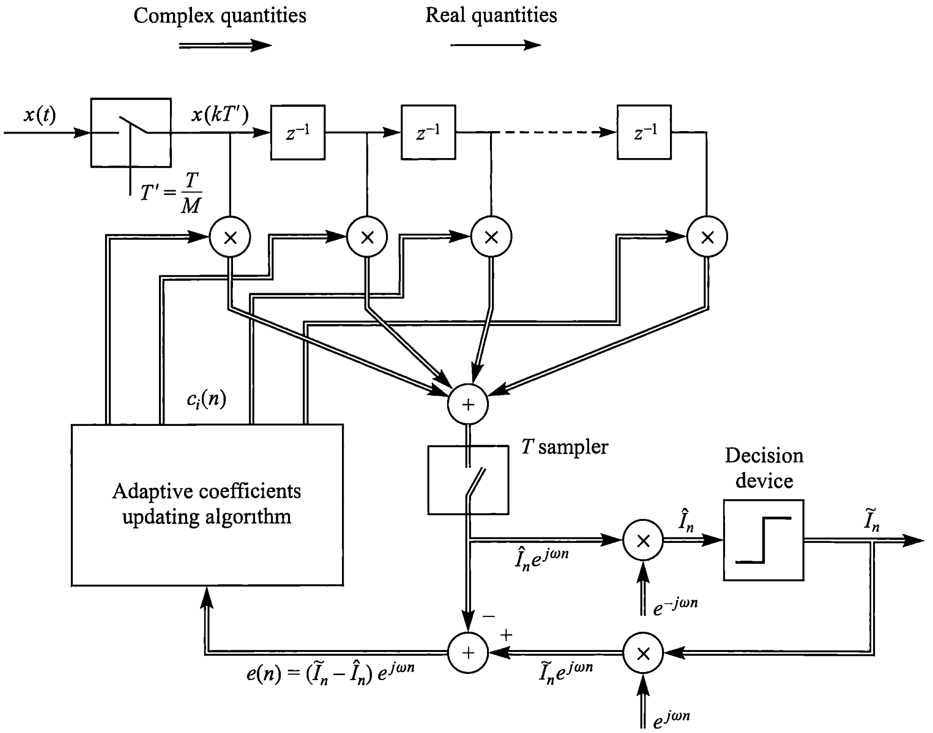

Complex quantities
Real quantities
$x(t)$
$x(kT')$
$T' = \frac{T}{M}$
z^{-1}
z^{-1}
z^{-1}
×
+
$c_i(n)$
Adaptive coefficients updating algorithm
T sampler
Decision device
$\hat{I}_n e^{j\omega n}$
$\hat{I}_n$
$\tilde{I}_n$
$e^{-j\omega n}$
−
+
$e(n) = (\tilde{I}_n - \hat{I}_n)\, e^{j\omega n}$
$\tilde{I}_n e^{j\omega n}$
$e^{j\omega n}$

FIGURE 9.4–12
Structure of a phase-splitting fractionally spaced equalizer. [*From Ling and Qureshi (1990); © 1990 IEEE.*]

9.5

DECISION-FEEDBACK EQUALIZATION

In Section 9.3–2 we developed an equivalent discrete-time model of the channel with ISI and additive noise, as shown in Figure 9.3–2. We observed that the additive Gaussian noise in this model is colored. Then we simplified this model by inserting a noise-whitening filter prior to the equalizer, so that the resulting discrete-time model of the channel has AWGN as shown in Figure 9.3–3. To recover the information sequence that is corrupted by ISI, we considered two types of equalization methods, one based on the MLSE criterion that is efficiently implemented by the Viterbi algorithm and the other employed a linear transversal filter. We recall that the MLSE is the optimum detector in the sense that it minimizes the probability of a sequence error while the linear equalizer is suboptimum.

In this section, we consider a nonlinear type of channel equalizer for mitigating the ISI, which is also suboptimum, but whose performance is generally better than that of the linear equalizer. The nonlinear equalizer consists of two filters, a feedforward filter and a feedback filter, arranged as shown in Figure 9.5–1, and it is called a *decision-feedback equalizer (DFE)*. The input to the feedforward filter is the received signal sequence. The feedback filter has as its input the sequence of decisions on previously detected symbols. Functionally, the feedback filter is used to remove that part of the ISI from the present estimated symbol caused by previously detected symbols. Since the detector feeds hard decisions to the feedback filter, the DFE is nonlinear.

In the case where the feedforward and feedback filters have infinite-duration impulse responses, Price (1972) showed that the optimum feedforward filter in a zero-forcing DFE is the noise-whitening filter with system function $1/F^*(1/z^*)$. Hence, in the zero-forcing DFE, the feedforward filter whitens the additive noise and results in an equivalent discrete-time channel having the system function $F(z)$.

In our treatment, we focus on finite-duration impulse response filters and apply the MSE criterion to optimize their coefficients.

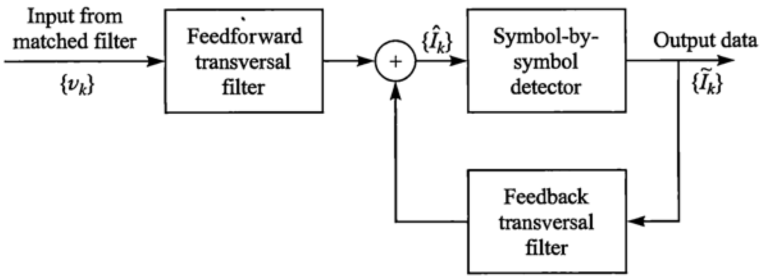

Input from matched filter
$\{v_k\}$
Feedforward transversal filter
+
$\{\hat{I}_k\}$
Symbol-by-symbol detector
Output data
$\{\tilde{I}_k\}$
Feedback transversal filter

FIGURE 9.5–1
Structure of decision-feedback equalizer.

9.5–1 Coefficient Optimization

From the description given above, it follows that the equalizer output can be expressed as

$$\hat{I}_k = \sum_{j=-K_1}^{0} c_j v_{k-j} + \sum_{j=1}^{K_2} c_j \tilde{I}_{k-j} \tag{9.5–1}$$

where $\hat{I}_k$ is an estimate of the kth information symbol, $\{c_j\}$ are the tap coefficients of the filter, and $\{\tilde{I}_{k-1}, \ldots, \tilde{I}_{k-K_2}\}$ are previously detected symbols. The equalizer is assumed to have $(K_1 + 1)$ taps in its feedforward section and K_2 in its feedback section.

Both the peak distortion criterion and the MSE criterion result in a mathematically tractable optimization of the equalizer coefficients, as can be concluded from the papers by George et al. (1971), Price (1972), Salz (1973), and Proakis (1975). Since the MSE criterion is more prevalent in practice, we focus our attention on it. Based on the assumption that previously detected symbols in the feedback filter are correct, the minimization of MSE

$$J(K_1, K_2) = E|I_k - \hat{I}_k|^2 \tag{9.5–2}$$

leads to the following set of linear equations for the coefficients of the feedforward filter:

$$\sum_{j=-K_1}^{0} \psi_{lj} c_j = f_{-l}^*, \qquad l = -K_1, \ldots, -1, 0 \tag{9.5–3}$$

where

$$\psi_{lj} = \sum_{m=0}^{-l} f_m^* f_{m+l-j} + N_0 \delta_{lj}, \qquad l, j = -K_1, \ldots, -1, 0 \tag{9.5–4}$$

The coefficients of the feedback filter of the equalizer are given in terms of the coefficients of the feedforward section by the following expression:

$$c_k = -\sum_{j=-K_1}^{0} c_j f_{k-j}, \qquad k = 1, 2, \ldots, K_2 \tag{9.5–5}$$

The values of the feedback coefficients result in complete elimination of intersymbol interference from previously detected symbols, provided that previous decisions are correct and that $K_2 \geq L$ (see Problem 9.51).

9.5–2 Performance Characteristics of DFE

We now turn our attention to the performance achieved with decision-feedback equalization. The exact evaluation of the performance is complicated to some extent by occasional incorrect decisions made by the detector, which then propagate down the feedback section. In the absence of decision errors, the minimum MSE is given as

$$J_{\min}(K_1) = 1 - \sum_{j=-K_1}^{0} c_j f_{-j} \tag{9.5–6}$$

By going to the limit ($K_1 \to \infty$) of an infinite number of taps in the feedforward filter, we obtain the smallest achievable MSE, denoted as $J_{\min}$. With some effort $J_{\min}$ can be expressed in terms of the spectral characteristics of the channel and additive noise, as shown by Salz (1973). This more desirable form for $J_{\min}$ is

$$J_{\min} = \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0}{X(e^{j\omega T}) + N_0}\right] d\omega\right\} \tag{9.5–7}$$

The corresponding output SNR is

$$\begin{aligned} \gamma_\infty &= \frac{1 - J_{\min}}{J_{\min}} \\ &= -1 + \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0 + X(e^{j\omega T})}{N_0}\right] d\omega\right\} \end{aligned} \tag{9.5–8}$$

We observe again, that in the absence of intersymbol interference, $X(e^{j\omega T}) = 1$, and hence, $J_{\min} = N_0/(1 + N_0)$. The corresponding output SNR is $\gamma_\infty = 1/N_0$.

EXAMPLE 9.5–1. It is interesting to compare the value of $J_{\min}$ for the decision-feedback equalizer with the value of $J_{\min}$ obtained with the linear MSE equalizer. For example, let us consider the discrete-time equivalent channel consisting of two taps f_0 and f_1. The minimum MSE for this channel is

$$\begin{aligned} J_{\min} &= \exp\left\{\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} \ln\left[\frac{N_0}{1 + N_0 + 2|f_0||f_1|\cos(\omega T + \theta)}\right] d\omega\right\} \\ &= N_0 \exp\left[-\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln(1 + N_0 + 2|f_0||f_1|\cos\omega)\, d\omega\right] \\ &= \frac{2N_0}{1 + N_0 + \sqrt{(1 + N_0)^2 - 4|f_0 f_1|^2}} \end{aligned} \tag{9.5–9}$$

Note that $J_{\min}$ is maximized when $|f_0| = |f_1| = \sqrt{\frac{1}{2}}$. Then

$$\begin{aligned} J_{\min} &= \frac{2N_0}{1 + N_0 + \sqrt{(1 + N_0)^2 - 1}} \\ &\approx 2N_0, \qquad N_0 \ll 1 \end{aligned} \tag{9.5–10}$$

The corresponding output SNR is

$$\gamma_\infty \approx \frac{1}{2N_0}, \qquad N_0 \ll 1 \tag{9.5–11}$$

Therefore, there is a 3-dB degradation in output SNR due to the presence of intersymbol interference. In comparison, the performance loss for the linear equalizer is very severe. Its output SNR as given by Equalizer 9.4–53 is $\gamma_\infty \approx (2/N_0)^{1/2}$ for $N_0 \ll 1$.

EXAMPLE 9.5–2. Consider the exponentially decaying channel characteristic of the form

$$f_k = (1 - a^2)^{1/2} a^k, \qquad k = 0, 1, 2, \ldots \tag{9.5–12}$$

where $a < 1$. The output SNR of the decision-feedback equalizer is

$$\begin{aligned}\gamma_\infty &= -1 + \exp\left\{\frac{1}{2\pi}\int_{-\pi}^{\pi} \ln\left[\frac{1 + a^2 + (1 - a^2)/N_0 - 2a\cos\omega}{1 + a^2 - 2a\cos\omega}\right] d\omega\right\} \\ &= -1 + \frac{1}{2N_0}\left\{1 - a^2 + N_0(1 + a^2) + \sqrt{[1 - a^2 + N_0(1 + a^2)]^2 - 4a^2N_0^2}\right\} \\ &\approx \frac{(1 - a^2)[1 + N_0(1 + a^2)/(1 - a^2)] - N_0}{N_0} \\ &\approx \frac{1 - a^2}{N_0}, \qquad N_0 \ll 1\end{aligned} \tag{9.5–13}$$

Thus, the loss in SNR is $10\log_{10}(1 - a^2)$ dB. In comparison, the linear equalizer has a loss of $10\log_{10}[(1 - a^2)/(1 + a^2)]$ dB.

These results illustrate the superiority of the decision-feedback equalizer over the linear equalizer when the effect of decision errors on performance is neglected. It is apparent that a considerable gain in performance can be achieved relative to the linear equalizer by the inclusion of the decision-feedback section, which eliminates the intersymbol interference from previously detected symbols.

One method of assessing the effect of decision errors on the error rate performance of the decision-feedback equalizer is Monte Carlo simulation on a digital computer. For purposes of illustration, we offer the following results for binary PAM signaling through the equivalent discrete-time channel models shown in Figure 9.4–5b and c.

The results of the simulation are displayed in Figure 9.5–2. First of all, a comparison of these results with those presented in Figure 9.4–4 leads us to conclude that the decision-feedback equalizer yields a significant improvement in performance relative to the linear equalizer having the same number of taps. Second, these results indicate that there is still a significant degradation in performance of the decision-feedback equalizer due to the residual intersymbol interference, especially on channels with severe distortion such as the one shown in Figure 9.4–5c. Finally, the performance loss due to incorrect decisions being fed back is 2 dB, approximately, for the channel responses under consideration. Additional results on the probability of error for a decision-feedback equalizer with error propagation may be found in the papers by Duttweiler et al. (1974) and Beaulieu (1994).

The structure of the DFE that is analyzed above employs a T-spaced filter for the feedforward section. The optimality of such a structure is based on the assumption that the analog filter preceding the DFE is matched to the channel-corrupted pulse response and its output is sampled at the optimum time instant. In practice, the channel response is not known a priori, so it is not possible to design an ideal matched filter. In view of this difficulty, it is customary in practical applications to use a fractionally spaced feedforward filter. Of course, the feedback filter tap spacing remains at T. The use of the FSE for the feedforward filter eliminates the system sensitivity to a timing error.

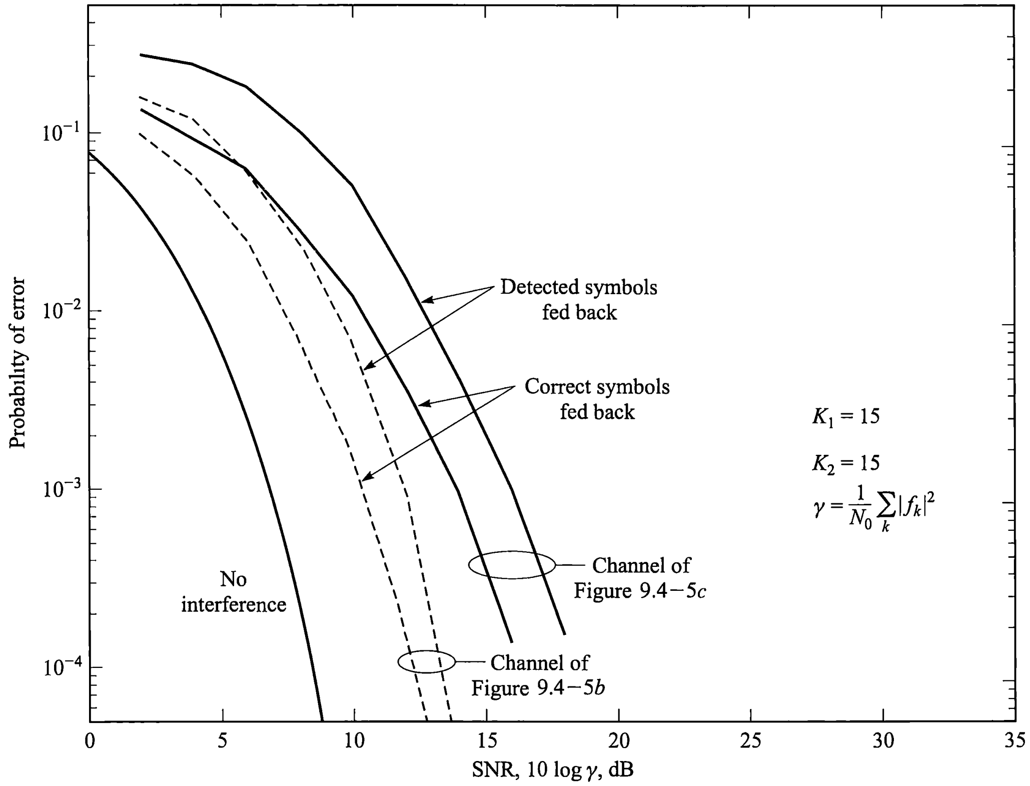
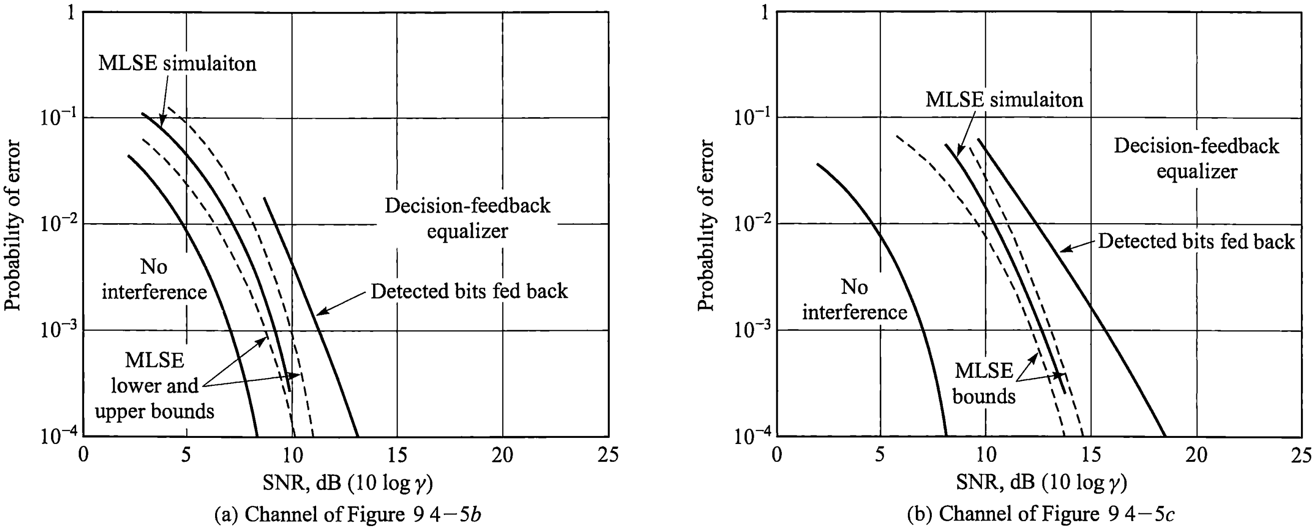


FIGURE 9.5–2
Performance of decision-feedback equalizer with and without error propagation.

Performance comparison with the MLSE We conclude this subsection on the performance of the DFE by comparing its performance against that of the MLSE. For the two-path channel with $f_0 = f_1 = \sqrt{\frac{1}{2}}$, we have shown that the MLSE suffers no SNR loss while the decision-feedback equalizer suffers a 3-dB loss. On channels with more distortion, the SNR advantage of the MLSE over decision-feedback equalization is even greater. Figure 9.5–3 illustrates a comparison of the error rate performance of these two equalization techniques, obtained via Monte Carlo simulation, for binary PAM and the channel characteristics shown in Figure 9.4–5b and c. The error rate curves for the two methods have different slopes; hence the difference in SNR increases as the error probability decreases. As a benchmark, the error rate for the AWGN channel with no intersymbol interference is also shown in Figure 9.5–3.

9.5–3 Predictive Decision-Feedback Equalizer

Belfiore and Park (1979) proposed another DFE structure that is equivalent to the one shown in Figure 9.5–1 under the condition that the feedforward filter has an infinite number of taps. This structure consists of an FSE as a feedforward filter and a linear predictor as a feedback filter, as shown in the configuration given in Figure 9.5–4. Let us briefly consider the performance characteristics of this equalizer, based on the MSE criterion.



(a) Channel of Figure 9 4−5*b*

(b) Channel of Figure 9 4−5*c*

FIGURE 9.5–3
Comparison of performance between MLSE and decision-feedback equalization for channel characteristics shown (a) in Figure 9.4–5b and (b) in Figure 9.4–5c.

First of all, the noise at the output of the infinite length feedforward filter has the power spectral density

$$\frac{N_0 X(e^{j\omega T})}{|N_0 + X(e^{j\omega T})|^2}, \qquad |\omega| \le \frac{\pi}{T} \tag{9.5–14}$$

The residual intersymbol interference has the power spectral density

$$\left|1 - \frac{X(e^{j\omega T})}{N_0 + X(e^{j\omega T})}\right|^2 = \frac{N_0^2}{|N_0 + X(e^{j\omega T})|^2}, \qquad |\omega| \le \frac{\pi}{T} \tag{9.5–15}$$

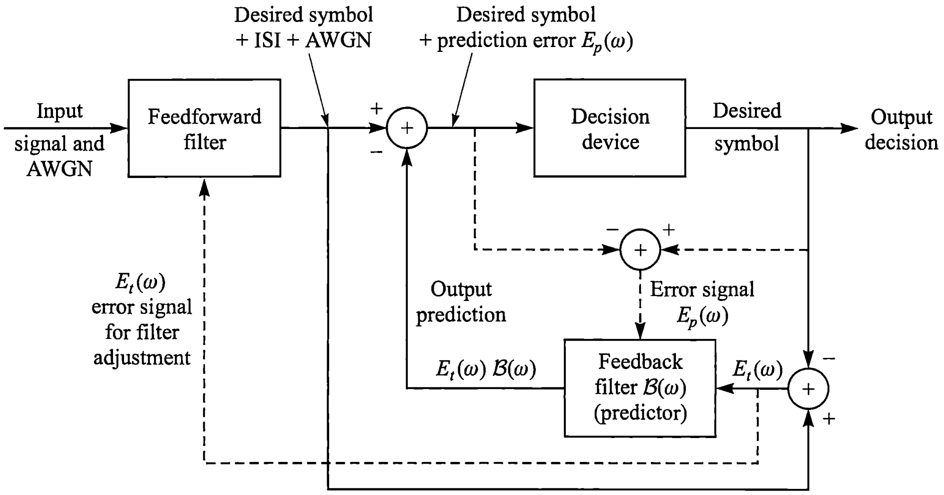

Desired symbol
+ ISI + AWGN
Desired symbol
+ prediction error $E_p(\omega)$
Input
signal and
AWGN
Feedforward filter
+
−
Decision device
Desired symbol
Output decision
−
+
Error signal
$E_p(\omega)$
Output prediction
$E_t(\omega)\,\mathcal{B}(\omega)$
Feedback filter $\mathcal{B}(\omega)$ (predictor)
$E_t(\omega)$
−
+
$E_t(\omega)$
error signal
for filter
adjustment

FIGURE 9.5–4
Block diagram of predictive DFE.

The sum of these two spectra represents the power spectral density of the total noise and intersymbol interference at the output of the feedforward filter. Thus, on adding Equations 9.5–14 and 9.5–15, we obtain

$$|E_t(\omega)|^2 = \frac{N_0}{N_0 + X(e^{j\omega T})}, \qquad |\omega| \le \frac{\pi}{T} \tag{9.5–16}$$

As we have observed previously, if $X(e^{j\omega T}) = 1$, the channel is ideal and, hence, it is not possible to reduce the MSE any further. On the other hand, if there is channel distortion, the power in the error sequence at the output of the feedforward filter can be reduced by means of linear prediction based on past values of the error sequence.

If $\mathcal{B}(\omega)$ represents the frequency response of the infinite length feedback predictor, i.e.,

$$\mathcal{B}(\omega) = \sum_{n=1}^{\infty} b_n e^{-j\omega nT} \tag{9.5–17}$$

then the error at the output of the predictor is

$$E_p(\omega) = E_t(\omega) - E_t(\omega)\mathcal{B}(\omega) = E_t(\omega)[1 - \mathcal{B}(\omega)] \tag{9.5–18}$$

The minimization of the mean square value of this error, i.e.,

$$J = \frac{1}{2\pi} \int_{-\pi/T}^{\pi/T} |1 - \mathcal{B}(\omega)|^2 |E_t(\omega)|^2 d\omega \tag{9.5–19}$$

over the predictor coefficients $\{b_n\}$ yields the optimum predictor in the form

$$\mathcal{B}(\omega) = 1 - \frac{G(\omega)}{g_0} \tag{9.5–20}$$

where $G(\omega)$ is the solution to the spectral factorization

$$G(\omega)G^*(-\omega) = \frac{1}{|E_t(\omega)|^2} \tag{9.5–21}$$

and

$$G(\omega) = \sum_{n=0}^{\infty} g_n e^{-j\omega nT} \tag{9.5–22}$$

The output of the infinite length linear predictor is a white noise sequence with power spectral density $1/g_0^2$ and the corresponding minimum MSE is given by Equation 9.5–7. Therefore, the MSE performance of the infinite length predictive DFE is identical to the conventional DFE.

Although these two DFE structures result in equivalent performance if their lengths are infinite, the predictive DFE is suboptimum if the lengths of the two filters are finite. The reason for the optimality of the conventional DFE is relatively simple. The optimization of its tap coefficients in the feedforward and feedback filters is done jointly. Hence, it yields the minimum MSE. On the other hand, the optimizations of the feedforward filter and the feedback predictor in the predictive DFE are done separately. Hence, its MSE is at least as large as that of the conventional DFE.

In spite of this suboptimality of the predictive DFE, it is suitable as an equalizer for trellis-coded signals, where the conventional DFE is not as suitable, as described in the next chapter.

9.5–4 Equalization at the Transmitter—Tomlinson–Harashima Precoding

If the channel response is known to the transmitter, the equalizer can be placed at the transmitter end of the communication system. Thus, the noise enhancement that is generally inherent when the equalizer (linear or DFE) is placed at the receiver is avoided. In practice, however, channel characteristics generally vary with time, so it is cumbersome to place the entire equalizer at the transmitter.

In wireline channels, the channel characteristics do not vary significantly with time. Therefore, it is possible to place the feedback filter of the DFE at the transmitter and the feedforward filter at the receiver. This approach has the advantage that the problem of error propagation due to incorrect decisions in the feedback filter is completely eliminated. Thus, the tail (postcursors) in the channel response is cancelled without any penalty in the SNR. The linear fractionally spaced feedforward part of the DFE, which ideally is the WMF, can be designed to compensate for ISI that results from any small time variation in the channel response. The synthesis of the feedback filter of the DFE at the transmitter side is usually performed after the response of the channel is measured at the receiver by the transmission of a channel probe signal and the receiver sends to the transmitter the coefficients of the feedback filter.

The one problem with this approach to implementing the DFE is that the signal points at the transmitter, after subtracting the postcursors of the ISI, generally have a larger dynamic range than the original signal constellation and, consequently, require a larger transmitter power. This problem can be avoided by precoding the information symbols prior to transmission as described by Tomlinson (1971) and Harashima and Miyakawa (1972).

We describe the precoding technique for a PAM signal constellation. Since a square QAM signal constellation may be viewed as two PAM signal sets on quadrature carriers, the precoding is easily extended to QAM. For simplicity, we assume that the feedforward filter in the DFE is the WMF and that the channel response, characterized by the parameters $\{f_i, 0 \le i \le L\}$, is perfectly known to the transmitter and the receiver. The information symbols $\{I_k\}$ are assumed to take the values $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$.

In the precoding, the ISI due to the postcursors $\{f_i, 1 \le i \le L\}$ is subtracted from the symbol to be transmitted and, if the difference falls outside of the range $(-M, M]$, it is reduced to the range by subtracting an integer multiple of $2M$ from this difference. Hence, the precoder output may be expressed as

$$a_k = I_k - \sum_{j=1}^{L} f_j a_{k-j} + 2Mb_k \tag{9.5–23}$$

where $\{b_k\}$ represents the appropriate integer that brings $\{a_k\}$ to the desired range. In other words, $\{a_k\}$ is reduced to the desired range by performing a modulo-$2M$ operation.

The modulo operation is defined mathematically by the function

$$m_y(x) = x - yz$$

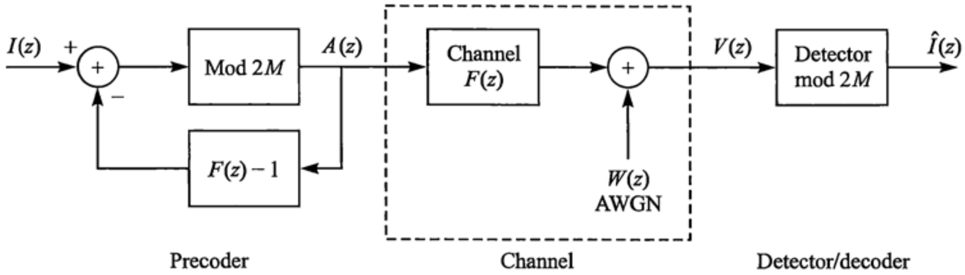


FIGURE 9.5–5
Tomlinson–Harashima precoding.

where $y > 0$ and $z = \left\lfloor \dfrac{x + y/2}{y} \right\rfloor$ is a unique integer such that $m_y(x)\,\epsilon[-y/2, y/2]$. In our case $y = 2M$. By using the z transform to describe the operation of the precoder, we have

$$A(z) = I(z) - [F(z) - 1]A(z) + 2MB(z) \tag{9.5–24}$$

where the channel coefficient f_0 is normalized to unity for convenience. Hence, the transmitted sequence is

$$A(z) = \frac{I(z) + 2MB(z)}{F(z)} \tag{9.5–25}$$

Since the channel response is $F(z)$, the received signal sequence may be expressed as

$$\begin{aligned} V(z) &= A(z) + W(z) \\ &= [I(z) + 2MB(z)] + W(z) \end{aligned} \tag{9.5–26}$$

where $W(z)$ represents the AWGN term. Therefore, the received data sequence term $I(z) + 2MB(z)$ at the input to the detector is free of ISI and $I(z)$ can be recovered from $V(z)$ by use of a symbol-by-symbol detector that decodes the symbols modulo-$2M$. Figure 9.5–5 illustrates the block diagram of the system that implements the precoder and the feedback filter of the DFE at the transmitter.

The placement of the feedback filter at the transmitter makes it possible to use the DFE in conjunction with trellis-coded modulation (TCM). Since the equalizer at the receiver is a linear filter, decisions from the output of the Viterbi (TCM) detector can be used to adjust the coefficients of the equalizer. In this case, the Viterbi detector performs the modulo-$2M$ operations in its metric computations.

9.6

REDUCED COMPLEXITY ML DETECTORS

The performance results of the three basic equalization methods described above, namely, MLSE, linear equalization (LE), and decision-feedback equalization (DFE), clearly show the superiority of MLSE in channels with severe ISI. Such channels are encountered in wireless communications and in high-density magnetic recording systems.

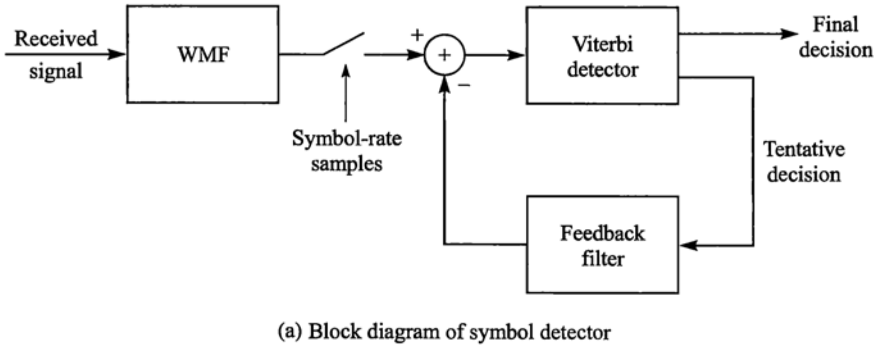
The performance advantage of MLSE has motivated a significant amount of research on methods that retain the performance characteristics of MLSE, but do so at a reduced complexity.

The early work on the design of reduced complexity MLSE focused on methods that reduce the length of the ISI span by preprocessing the received signal prior to the maximum-likelihood detector. Falconer and Magee (1973) and Beare (1978) used a linear equalizer to reduce the span of the ISI to some small specified length prior to the Viterbi detector. Lee and Hill (1977) employed a DFE in place of the LE. Thus, the large ISI span in the channel is reduced to a sufficiently small length, called the *desired impulse response*, so that the complexity of the Viterbi detector following the LE or DFE is manageable. We may view this role of the LE or the DFE, prior to the Viterbi detector, as equalizing the channel response to a specified partial-response characteristic of short duration (the desired impulse response) which the Viterbi detector can handle with sufficiently small complexity. The choice of the desired impulse response is tailored to the ISI characteristics of the channel. This approach to reducing the complexity of the Viterbi detector has proved to be very effective in high-density magnetic recording systems, as illustrated in the papers by Siegel and Wolf (1991), Tyner and Proakis (1993), Moon and Carley (1988), and Proakis (1998).

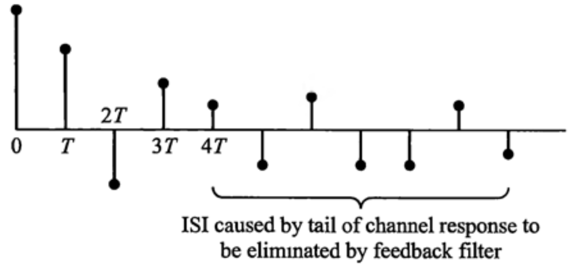
Another general approach is to reduce the complexity of the Viterbi detector directly, by reducing the number of surviving sequences. The papers by Vermuelen and Hellman (1974), Fredricsson (1974), and Foschini (1977) describe algorithms that reduce the number of surviving sequences in the Viterbi detector. Other works on this class of methods include the papers by Clark et al. (1984, 1985) and Wesolowski (1987a).

The most effective approach in terms of performance for reducing the complexity of the Viterbi detector directly is the method described in the papers by Bergmans et al. (1987), Eyuboglu and Qureshi (1988), and Duel-Hallen and Heegard (1989). The filter preceding the Viterbi detector is the whitened matched filter (WMF) described previously. The WMF reduces the channel to one that has a minimum phase characteristic. The basic algorithm described in these papers for reducing the computational complexity of the Viterbi detector employs decision feedback within the Viterbi detector to reduce the effective length of the ISI from L terms to L_0 terms, where $L_0 < L$. This may be accomplished in one of two ways, as described by Bergmans et al. (1987), either by using "global feedback" or "local feedback" from preliminary decisions that are present in the Viterbi detector. The use of global feedback is illustrated in Figure 9.6–1, where preliminary decisions obtained by using the most probable surviving sequence from the Viterbi detector are used to synthesize the tail in the ISI due to the channel coefficients $(f_{L_0+1}, f_{L_0+2}, \ldots, f_{L-1}, f_L)$. Thus, for M-ary modulations, the computational complexity of the Viterbi detector is reduced from M^L to M^{L_0}, which amounts to a reduction by the factor M^{L-L_0}. The primary drawback of using global feedback is that if one or more of the symbols $\hat{I}_{k-L_0-1}, \ldots, \hat{I}_{k-L}$ in the most probable surviving sequence are incorrect, the subtraction of the tail in the ISI is also incorrect and, thus, the metric computations are corrupted by the residual ISI resulting from this imperfect cancellation.

To remedy this problem, one may use the preliminary decisions corresponding to each surviving sequence to cancel the ISI in the tail of the corresponding surviving sequence. Thus, the ISI will be perfectly cancelled when the correct sequence is



(a) Block diagram of symbol detector



(b) Channel response

FIGURE 9.6–1
Reduced complexity ML sequence detector using feedback from the Viterbi detector.

among the surviving sequences, even if it is not the most probable sequence. Bergmans et al. (1987) described this approach as using "local feedback" to perform the tail cancellation.

It is interesting to note that if L_0 is selected as unity ($L_0 = 1$), the Viterbi detector reduces to the simple feedback filter of a conventional DFE. At the other extreme, when $L_0 = L$, we have a full complexity Viterbi detector. The analytical and simulation results given in the paper by Bergmans et al. (1987) clearly illustrate that local feedback gives superior performance to global feedback.

9.7
ITERATIVE EQUALIZATION AND DECODING—TURBO EQUALIZATION

Iterative decoding and the turbo-coding principle that was described in Section 8.7 can be applied to channel equalization. Suppose the transmitter of a digital communication system employs a binary systematic convolutional encoder followed by a block interleaver and a modulator. The channel is a linear time-dispersive channel that introduces ISI. In such a case, we may view the channel as an inner encoder in a serially concatenated code. Hence, we can apply iterative decoding based on the MAP criterion.

The basic configuration of the iterative equalizer–decoder is shown in Figure 9.7–1. The input to the MAP equalizer is the sequence $\{v_k\}$ from the WMF. The equalizer computes the logarithm of the likelihood ratio of the coded bits, denoted as

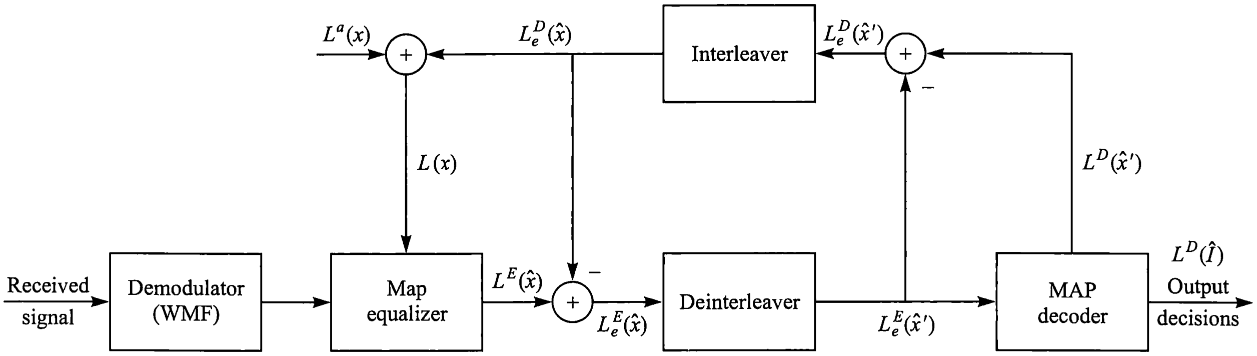


FIGURE 9.7–1
Iterative equalization and decoding.

$L^E(\hat{x})$, which represents the a posteriori values of the coded bits. The outer decoder receives as an input the extrinsic part of $L^E(\hat{x})$, which is defined as

$$L_e^E(\hat{x}) = L^E(\hat{x}) - L_e^D(\hat{x}) \tag{9.7–1}$$

where $L_e^D(\hat{x})$ is the extrinsic part of the outer decoder output after interleaving. $L_e^E(\hat{x})$ is deinterleaved prior to being fed to the outer decoder.

The outer decoder computes the logarithm of the likelihood ratio of the coded bits, denoted by $L^D(\hat{x}')$ and the information bits, denoted as $L^D(\hat{I})$. The extrinsic part of $L^D(\hat{x}')$, denoted as $L_e^D(\hat{x}')$, is the incremental information about the current bit obtained by the decoder after observing all the information for all the received bits. The extrinsic information is computed as

$$L_e^D(\hat{x}') = L^D(\hat{x}') - L_e^E(\hat{x}') \tag{9.7–2}$$

$L_e^D(\hat{x}')$ is interleaved to produce $L_e^D(\hat{x})$ and fed to the MAP equalizer. We emphasize the importance of feeding back only the extrinsic part $L_e^D(\hat{x})$, thus, minimizing the correlation between the a priori information used by the equalizer and previous equalizer outputs. Similarly, we reduce the a posteriori information $L^E(\hat{x})$ by the a priori information values $L_e^D(\hat{x})$ to obtain the extrinsic information value $L_e^E(\hat{x})$, which is fed to the outer decoder after deinterleaving.

The computation of the log-likelihood ratios is described in the paper by Bauch et al. (1997). The power of this iterative equalization–decoding scheme can be assessed from the performance results given in this paper. Figure 9.7–2 illustrates the bit error probability obtained through simulation of the five-tap time-invariant channel given in Figure 9.4–5c. The outer decoder used is a rate 1/2 recursive systematic convolutional code with constraint length $K = 5$. The interleaver used was a pseudorandom block interleaver of length $N = 4096$ bits. Binary PSK was used for modulation. The graph illustrates the performance gain as the number of iterations is increased. We observe that after six iterations, the performance of the iterative equalizer–decoder is within 0.8 dB of the performance of the encoded data without ISI, at a bit error probability of 10^{-4}. Hence, the iterative equalizer eliminates nearly the entire loss due to ISI. In contrast, the optimum (noniterative) Viterbi detector for this channel suffers a loss of approximately 7 dB, due to ISI, as can be observed from Figure 9.5–3b. Therefore,

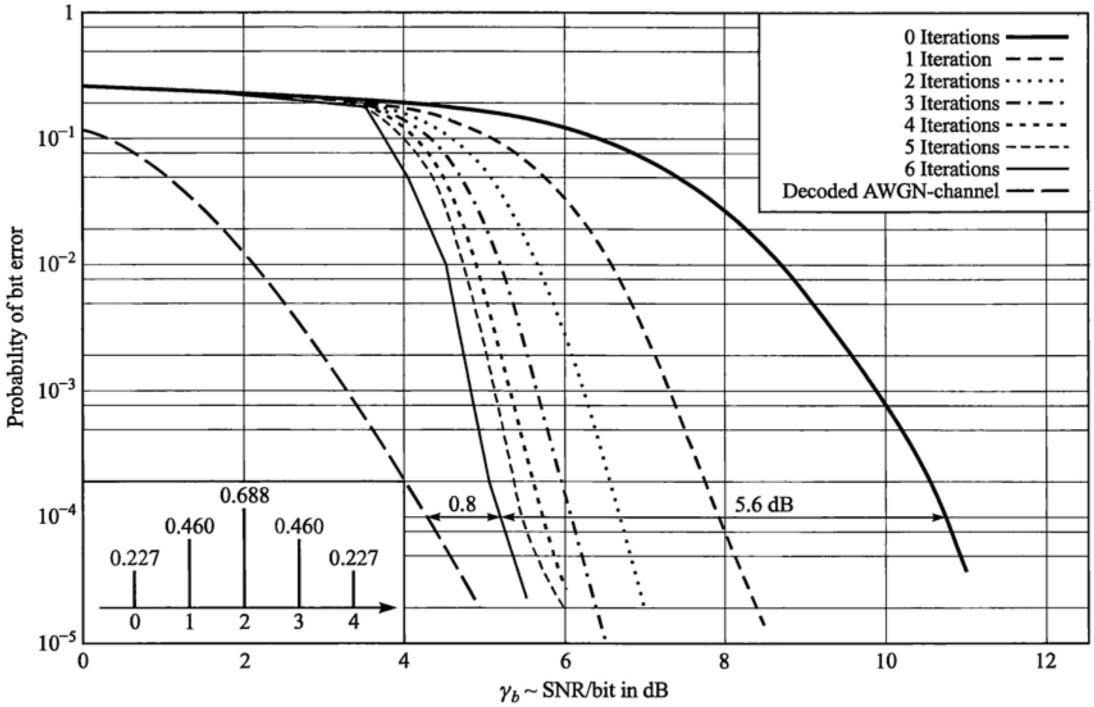


FIGURE 9.7–2
Channel taps and bit error rate for a time-invariant channel. [*From Bauch et al. (1997).*]

the iterative equalizer has achieved a performance gain of about 6 dB, aside from the coding gain due to the convolutional code. The performance of this method of iterative equalization has been evaluated for cellular radio channels by Bauch et al. (1998). An implementation of iterative equalization–decoding using non-linear circuits is described in a paper by Hagenauer et al. (1999).

An alternative approach to iterative equalization–decoding is to employ a parallel concatenated code (turbo code) followed by a block interleaver and a modulator at the transmitter side. The receiver employs a MAP equalizer followed by a turbo decoder. The extrinsic information generated by the turbo decoder is fed back to the MAP equalizer. Thus, we have an iterative equalizer–turbo decoder structure, which is called a turbo equalizer. Turbo equalization is treated by Raphaeli and Zarai (1998) and Douillard et al. (1995).

9.8

BIBLIOGRAPHICAL NOTES AND REFERENCES

The pioneering work on signal design for bandwidth-constrained channels was done by Nyquist (1928). The use of binary partial-response signals was originally proposed by Lender (1963) and was later generalized by Kretzmer (1966). Other early work on problems dealing with intersymbol interference (ISI) and transmitter and receiver optimization with constraints on ISI was done by Gerst and Diamond (1961), Tufts (1965), Smith (1965), and Berger and Tufts (1967). "Faster than Nyquist" transmission has been studied by Mazo (1975) and Foschini (1984).

Channel equalization for digital communications was developed by Lucky (1965, 1966), who focused on linear equalizers that were optimized using the peak distortion criterion. The mean-square-error criterion for optimization of the equalizer coefficients was proposed by Widrow (1966).

Decision-feedback equalization was proposed and analyzed by Austin (1967). Analyses of the performance of the DFE can be found in the papers by Monsen (1971), George et al. (1971), Price (1972), Salz (1973), Duttweiler et al. (1974), and Altekar and Beaulieu (1993).

The use of the Viterbi algorithm as the optimal maximum-likelihood sequence estimator for symbols corrupted by ISI was proposed and analyzed by Forney (1972) and Omura (1971). Its use for carrier-modulated signals was considered by Ungerboeck (1974) and MacKenchnie (1973).

The use of iterative MAP algorithms in suppressing ISI in coded systems, called turbo equalization, represents a major new advance in suppression of intersymbol interference in signal transmission through band-limited channels. It is anticipated that iterative MAP equalization algorithms will be incorporated in future communication systems. The implementation of turbo equalization, described in the paper by Hagenauer et al. (1999), is the first attempt at implementing an iterative MAP equalization algorithm in a coded system.

PROBLEMS

9.1 A channel is said to be *distortionless* if the response $y(t)$ to an input $x(t)$ is $Kx(t-t_0)$, where K and t_0 are contants. Show that if the frequency response of the channel is $A(f)e^{j\theta(f)}$, where $A(f)$ and $\theta(f)$ are real, the necessary and sufficient conditions for distortionless transmission are $A(f) = K$ and $\theta(f) = 2\pi f t_0 \pm n\pi, n = 0, 1, 2, \ldots$.

9.2 The raised cosine spectral characteristic is given by Equation 9.2–26.

a. Show that the corresponding impulse response is

$$x(t) = \frac{\sin(\pi t/T)}{\pi t/T} \frac{\cos(\beta\pi t/T)}{1 - 4\beta^2 t^2/T^2}$$

b. Determine the Hilbert transform of $x(t)$ when $\beta = 1$.

c. Does $\hat{x}(t)$ possess the desirable properties of $x(t)$ that make it appropriate for data transmission? Explain.

d. Determine the envelope of the SSB suppressed-carrier signal generated from $x(t)$.

9.3 *a.* Show that (Poisson sum formula)

$$x(t) = \sum_{k=-\infty}^{\infty} g(t)h(t-kT) \Rightarrow X(f) = \frac{1}{T}\sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right) G\left(f - \frac{n}{T}\right)$$

Hint: Make a Fourier-series expansion of the periodic factor

$$\sum_{k=-\infty}^{\infty} h(t-kT)$$

b. Using the result in (a), verify the following versions of the Poisson sum:

$$\sum_{k=-\infty}^{\infty} h(kT) = \frac{1}{T}\sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right) \qquad \text{(i)}$$

$$\sum_{k=-\infty}^{\infty} h(t-kT) = \frac{1}{T}\sum_{n=-\infty}^{\infty} H\left(\frac{n}{T}\right)\exp\left(\frac{j2\pi nt}{T}\right) \qquad \text{(ii)}$$

$$\sum_{k=-\infty}^{\infty} h(kT)\exp(-j2\pi kTf) = \frac{1}{T}\sum_{n=-\infty}^{\infty} H\left(f-\frac{n}{T}\right) \qquad \text{(iii)}$$

c. Derive the condition for no intersymbol interference (Nyquist criterion) by using the Poisson sum formula.

9.4 Suppose a digital communication system employs Gaussian-shaped pulses of the form

$$x(t) = \exp(-\pi a^2 t^2)$$

To reduce the level of intersymbol interference to a relatively small amount, we impose the condition that $x(T) = 0.01$, where T is the symbol interval. The bandwidth W of the pulse $x(t)$ is defined as that value of W for which $X(W)/X(0) = 0.01$, where $X(f)$ is the Fourier transform of $x(t)$. Determine the value of W and compare this value to that of raised cosine spectrum with 100 percent rolloff.

9.5 Show that the impulse response of a filter having a square-root raised cosine spectral characteristic is given as

$$x_{sr}(t) = \frac{(4\beta t/T)\cos[\pi(1+\beta)t/T] + \sin[\pi(1-\beta)t/T]}{(\pi t/T)[1-(4\beta t/T)^2]}$$

9.6 It is desired to implement a (discrete-time) finite impulse response (FIR) filter that provides square-root raised cosine spectral shaping. The coefficients of the FIR filter are the sampled values of the time response given in Problem 9.5, where the samples are taken at $t = kT/2$, for $k = 0, \pm 1, \pm 2, \cdots, \pm N$.

a. Determine the effect on the spectral characteristic resulting from the truncation of the filter response for $N = 10$, 15, and 20 and roll-off factor $\beta = 1/2$, by computing their frequency response

$$X_{sr}(\omega) = \sum_{n=-N}^{N} x(nT_s)e^{-j\omega n T_s}$$

where $T_s = T/2$.

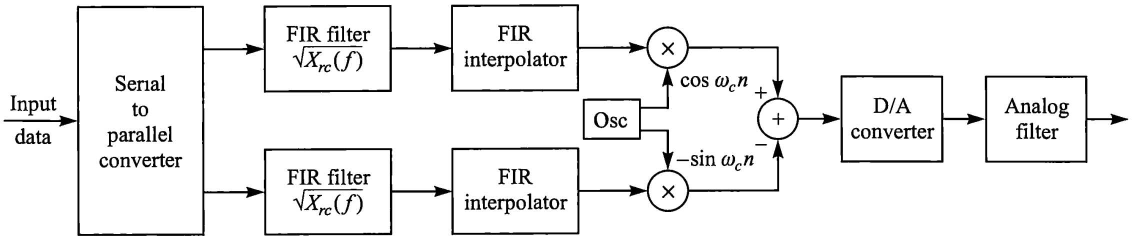
b. Plot the spectral characteristics of these three filters for $N = 10$, 15, and 20 and compare your results with the ideal square-root raised cosine spectrum.

9.7 Figure P9.7 illustrates a block diagram of a QAM or PSK modulator and demodulator (modem) in which the modulated signals are synthesized digitally and demodulated digitally. The FIR filters have square-root raised cosine spectral characteristics and employ a sampling rate of $2/T$, where the symbol rate $1/T = 2400$ symbols/s. The FIR interpolators employ a sampling rate of $6/T$ and are designed as linear phase FIR filters that pass the desired signal spectrum.

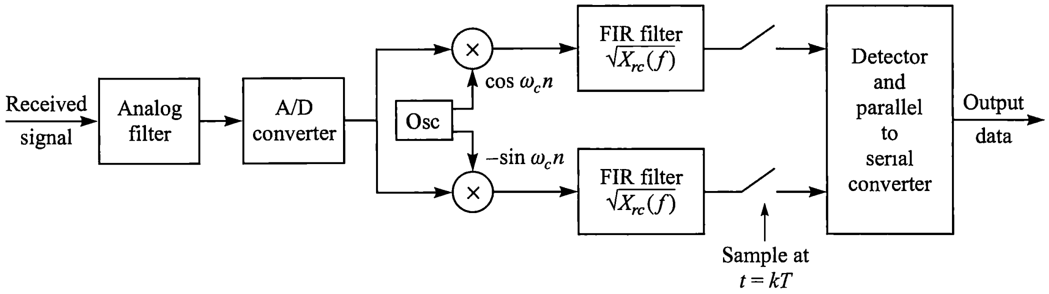
a. Write a software program that implements the digital modulator in Figure P9.7 for the following parameters: roll-off factor $\beta = 0.25$, length of FIR shaping filter $= 21$, length of FIR interpolator $= 11$, carrier frequency $f_c = 1800$ Hz.

b. Generate 5000 samples of the digital signal sequence $x_d(n)$ and compute and plot the power spectral density of this modulated signal.

c. Repeat (b) for five more iterations and compute the average power spectrum over the total of six signal records. Comment on the results.


Input data
Serial to parallel converter
FIR filter $\sqrt{X_{rc}(f)}$
FIR interpolator
FIR filter $\sqrt{X_{rc}(f)}$
FIR interpolator
×
×
Osc
$\cos \omega_c n$
$-\sin \omega_c n$
+
+
−
D/A converter
Analog filter

(a) QAM or PSK modulator


Received signal
Analog filter
A/D converter
Osc
×
×
$\cos \omega_c n$
$-\sin \omega_c n$
FIR filter $\sqrt{X_{rc}(f)}$
FIR filter $\sqrt{X_{rc}(f)}$
Sample at $t = kT$
Detector and parallel to serial converter
Output data

(b) QAM or PSK demodulator

FIGURE P9.7

9.8 (Carrierless QAM or PSK modem) Consider the transmission of a QAM or M-ary PSK ($M \geq 4$) signal at a carrier frequency f_c, where the carrier is comparable to the bandwidth of the baseband signal. The bandpass signal may be represented as

$$s(t) = \text{Re}\left[\sum_n I_n g(t - nT) e^{j2\pi f_c t}\right]$$

a. Show that $s(t)$ can be expressed as

$$s(t) = \text{Re}\left[\sum_n I'_n Q(t - nT)\right]$$

where $Q(t)$ is defined as

$$Q(t) = q(t) + j\hat{q}(t)$$
$$q(t) = g(t)\cos 2\pi f_c t$$
$$\hat{q}(t) = g(t)\sin 2\pi f_c t$$

and I'_n is a phase rotated symbol, i.e., $I'_n = I_n e^{j2\pi f_c nT}$.

b. Using FIR filters with responses $q(t)$ and $\hat{q}(t)$, sketch the block diagram of the modulator and demodulator implementation that does not require the mixer to translate the signal to bandpass at the modulator and to baseband at the demodulator.

9.9 (Carrierless amplitude or phase [CAP] modulation) In some practical applications in wireline data transmission, the bandwidth of the signal to be transmitted is comparable to the carrier frequency. In such systems, it is possible to eliminate the step of mixing the baseband signal with the carrier component. Instead, the bandpass signal can be synthesized directly, by embedding the carrier component in the realization of the FIR shaping filters. Thus, the modem is realized as shown in the block diagram in Figure P9.9, where the FIR shaping filters have the impulse responses

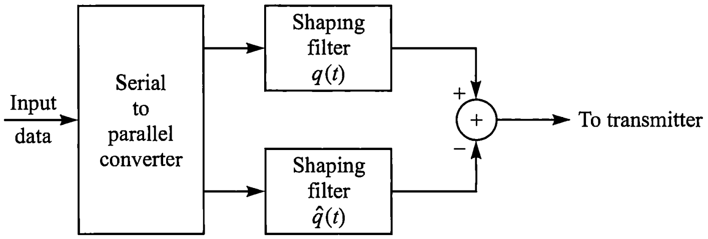
$$q(t) = g(t)\cos 2\pi f_c t$$
$$\hat{q}(t) = g(t)\sin 2\pi f_c t$$

and $g(t)$ is a pulse that has a square-root raised cosine spectral characteristic.

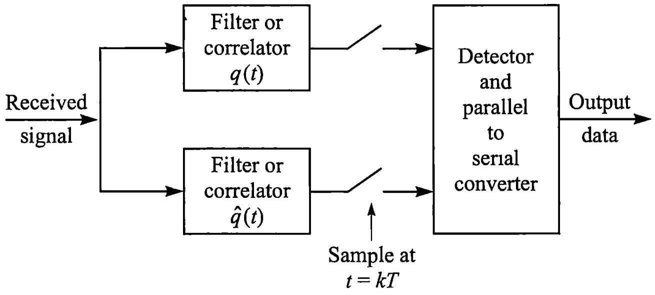
a. Show that

$$\int_{-\infty}^{\infty} q(t)\hat{q}(t)\,dt = 0$$

and that this system can be used to transmit two-dimensional signal constellations.


Input data
Serial to parallel converter
Shaping filter $q(t)$
Shaping filter $\hat{q}(t)$
+
−
+
To transmitter

(a) Modulator


Received signal
Filter or correlator $q(t)$
Filter or correlator $\hat{q}(t)$
Sample at $t = kT$
Detector and parallel to serial converter
Output data

(b) Demodulator

FIGURE P9.9

b. Under what conditions is this CAP modem identical to the carrierless QAM/PSK modem treated in Problem 9.8.

9.10 A band-limited signal having bandwidth W can be represented as

$$x(t) = \sum_{n=-\infty}^{\infty} x_n \frac{\sin[2\pi W(t - n/2W)]}{2\pi W(t - n/2W)}$$

a. Determine the spectrum $X(f)$ and plot $|X(f)|$ for the following cases:

$$x_0 = 2, \quad x_1 = 1, \quad x_2 = -1, \quad x_n = 0, \quad n \neq 0, 1, 2 \qquad \text{(i)}$$

$$x_{-1} = -1, \quad x_0 = 2, \quad x_1 = -1, \quad x_n = 0, \quad n \neq -1, 0, 1 \qquad \text{(ii)}$$

b. Plot $x(t)$ for these two cases.

c. If these signals are used for binary signal transmission, determine the number of received levels possible at the sampling instants $t = nT = n/2W$ and the probabilities of occurrence of the received levels. Assume that the binary digits at the transmitter are equally probable.

9.11 A 4-kHz bandpass channel is to be used for transmission of data at a rate of 9600 bits/s. If $\frac{1}{2}N_0 = 10^{-10}$ W/Hz is the spectral density of the additive zero-mean Gaussian noise in the channel, design a QAM modulation and determine the average power that achieves a bit error probability of 10^{-6}. Use a signal pulse with a raised cosine spectrum having a roll-off factor of at least 50 percent.

9.12 Determine the bit rate that can be transmitted through a 4-kHz voice-band telephone (bandpass) channel if the following modulation methods are used:

a. Binary PAM.
b. Four-phase PSK.
c. 8-point QAM.
d. Binary orthogonal FSK, with noncoherent detection.
e. Orthogonal four-FSK with noncoherent detection.
f. Orthogonal 8-FSK with noncoherent detection.

For (*a*)–(*c*), assume that the transmitter pulse shape has a raised cosine spectrum with a 50 percent roll-off.

9.13 An ideal voice-band telephone line channel has a band-pass frequency-response characteristic spanning the frequency range 600–3000 Hz.

a. Design an $M = 4$ PSK (quadrature PSK or QPSK) system for transmitting data at a rate of 2400 bits/s and a carrier frequency $f_c = 1800$ Hz. For spectral shaping, use a raised cosine frequency-response characteristic. Sketch a block diagram of the system and describe the functional operation of each block.

b. Repeat (*a*) for a bit rate $R = 4800$ bits/s and a 8-QAM signal.

9.14 A voice-band telephone channel passes the frequencies in the band from 300 to 3300 Hz. It is desired to design a modem that transmits at a symbol rate of 2400 symbols/s, with the objective of achieving 9600 bits/s. Select an appropriate QAM signal constellation, carrier frequency, and the roll-off factor of a pulse with a raised cosine spectrum that utilizes the entire frequency band. Sketch the spectrum of the transmitted signal pulse and indicate the important frequencies.

9.15 A communication system for a voice-band (3 kHz) channel is designed for a received SNR at the detector of 30 dB when the transmitter power is $P_s = -3$ dBW. Determine the value of P_s if it is desired to expand the bandwidth of the system to 10 kHz, while maintaining the same SNR at the detector.

9.16 Show that a pulse having the raised-cosine spectrum given by Equation 9.2–26 satisfies the Nyquist criterion given by Equation 9.2–13 for any value of the roll-off factor β.

9.17 Show that, for any value of β, the raised cosine spectrum given by Equation 9.2–26 satisfies

$$\int_{-\infty}^{\infty} X_{rc}(f)\,df = 1$$

[*Hint*: Use the fact that $X_{rc}(f)$ satisfies the Nyquist criterion given by Equation 9.2–13.]

9.18 The Nyquist criterion gives the necessary and sufficient condition for the spectrum $X(f)$ of the pulse $x(t)$ that yields zero ISI. Prove that for any pulse that is band-limited to $|f| < 1/T$, the zero-ISI condition is satisfied if Re$[X(f)$, for $f > 0$, consists of a rectangular function plus an arbitrary odd function around $f = 1/2T$, and Im$[X(f)]$ is any arbitrary even function around $f = 1/2T$.

9.19 A voice-band telephone channel has a passband characteristic in the frequency range $300\text{ Hz} < f < 3000\text{ Hz}$.

a. Select a symbol rate and a power efficient constellation size to achieve 9600 bits/s signal transmission.

b. If a square-root raised cosine pulse is used for the transmitter pulse $g(t)$, select the roll-off factor. Assume that the channel has an ideal frequency-response characteristic.

9.20 Design an M-ary PAM system that transmits digital information over an ideal channel with bandwidth $W = 2400$ Hz. The bit rate is 14,400 bits/s. Specify the number of transmitted points, the number of received signal points using a duobinary signal pulse, and the required $\mathcal{E}_b$ to achieve an error probability of 10^{-6}. The additive noise is zero-mean Gaussian with a power spectral density of 10^{-4}W/Hz.

9.21 A binary PAM signal is generated by exciting a raised cosine roll-off filter with a 50 percent roll-off factor and is then DSB/SC amplitude-modulated on a sinusoidal carrier as illustrated in Figure P9.21. The bit rate is 2400 bits/s.

a. Determine the spectrum of the modulated binary PAM signal and sketch it.

b. Draw the block diagram illustrating the optimum demodulator/detector for the received signal, which is equal to the transmitted signal plus additive white Gaussian noise.

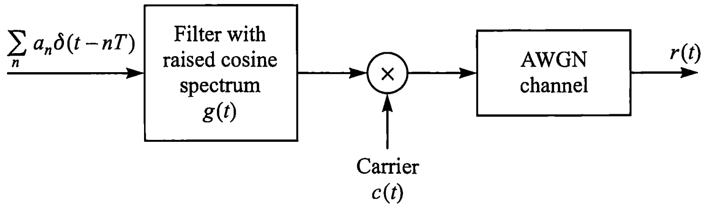


FIGURE P9.21

9.22 The elements of the sequence $\{a_n\}_{n=-\infty}^{\infty}$ are independent binary random variables taking values of ±1 with equal probability. This data sequence is used to modulate the basic pulse $g(t)$ shown in Figure P9.22a. The modulated signal is

$$X(t) = \sum_{n=-\infty}^{+\infty} a_n g(t - nT)$$

a. Find the power spectral density of $X(t)$.

b. If $g_1(t)$ (shown in Figure 9.22b) is used instead of $g(t)$, how would the power spectrum in (a) change?

c. In (b) assume we want to have a null in the spectrum at $f = 1/3T$. This is done by a precoding of the form $b_n = a_n + \alpha a_{n-3}$. Find the α that provides the desired null.

d. Is it possible to employ a precoding of the form $b_n = a_n + \sum_{i=1}^{N} \alpha_i a_{n-i}$ for some finite N such that the final power spectrum will be identical to zero for $1/3T \le |f| \le 1/2T$? If yes, how? If no, why? [*Hint*: Use properties of analytic functions.]

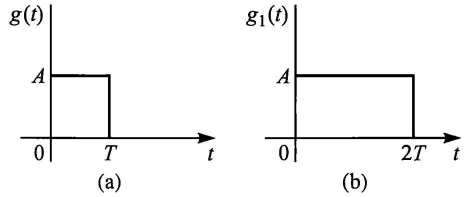


FIGURE P9.22

9.23 Consider the transmission of data via PAM over a voice-band telephone channel that has a bandwidth of 3000 Hz. Show how the symbol rate varies as a function of the excess bandwidth. In particular, determine the symbol rate for an excess bandwidth of 25, 33, 50, 67, 75 and 100 percent.

9.24 The binary sequence 10010110010 is the input to a precoder whose output is used to modulate a duobinary transmitting filter. Construct a table as in Table 9.2–1 showing the precoded sequence, the transmitted amplitude levels, the received signal levels, and the decoded sequence.

9.25 Repeat Problem 9.24 for a modified duobinary signal pulse.

9.26 A precoder for a partial response signal fails to work if the desired partial response at $n = 0$ is zero modulo M. For example, consider the desired response for $M = 2$:

$$x(nT) = \begin{cases} 2 & (n = 0) \\ 1 & (n = 1) \\ -1 & (n = 2) \\ 0 & (\text{otherwise}) \end{cases}$$

Show why this response cannot be precoded.

9.27 Consider the RC low-pass filter shown in Figure P9.27, where $\tau = RC = 10^{-6}$.

a. Determine and sketch the envelope (group) delay of the filter as a function of frequency.

b. Suppose that the input to the filter is a lowpass signal of bandwidth $\Delta f = 1$ kHz. Determine the effect of the RC filter on this signal.

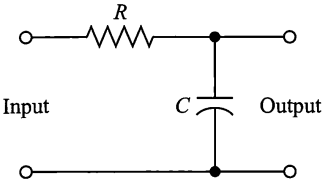


FIGURE P9.27

9.28 A microwave radio channel has a frequency response

$$C(f) = 1 + 0.3\cos 2\pi f T$$

Determine the frequency-response characteristic of the transmitting and receiving filters that yield zero ISI at a rate of $1/T$ symbols/s and have a 50 percent excess bandwidth. Assume that the additive noise spectrum is flat.

9.29 $M = 4$ PAM modulation is used for transmitting at a bit rate of 9600 bits/s on a channel having a frequency response

$$C(f) = \frac{1}{1 + j(f/2400)}$$

for $|f| \le 2400$, and $C(f) = 0$ otherwise. The additive noise is zero-mean white Gaussian with power spectral density $\frac{1}{2}N_0$ W/Hz. Determine the (magnitude) frequency-response characteristic of the optimum transmitting and receiving filters.

9.30 Use the Cauchy–Schwarz inequality to show that the transmitter and receiver filters given by Equation 9.2–83 minimize the noise-to signal ratio σ_ν^2/d^2, where σ_ν^2 is the noise power given by Equation 9.2–77, where $\mathcal{S}_{nn}(f) = N_0/2$.

9.31 Suppose that a channel frequency response is given as

$$C(f) = \begin{cases} 1 & |f| \le W/2 \\ \frac{1}{2} & \dfrac{W}{2} < |f| < W \end{cases}$$

Determine the loss in SNR incurred, as given by Equations 9.2–87 and 9.2–88, for the filters given by the corresponding Equations 9.2–79 and 9.2–83, respectively. Which filters result in a smaller loss?

9.32 In a binary PAM system, the input to the detector is

$$y_m = a_m + n_m + i_m$$

where $a_m = \pm 1$ is the desired signal, n_m is a zero-mean Gaussian random variable with variance σ_n^2, and i_m represents the ISI due to channel distortion. The ISI term is a random variable that takes the values $-\frac{1}{2}$, 0, and $\frac{1}{2}$ with probabilities $\frac{1}{4}$, $\frac{1}{2}$, and $\frac{1}{4}$, respectively. Determine the average probability of error as a function of σ_n^2.

9.33 In a binary PAM system, the clock that specifies the sampling of the correlator output is offset from the optimum sampling time by 10 percent.

a. If the signal pulse used is rectangular, determine the loss in SNR due to the mistiming.

b. Determine the amount of ISI introduced by the mistiming and determine its effect on performance.

9.34 The frequency-response characteristic of a lowpass channel can be approximated by

$$H(f) = \begin{cases} 1 + \alpha \cos 2\pi f t_0 & |\alpha| < 1, |f| \le W \\ 0 & \text{otherwise} \end{cases}$$

where W is the channel bandwidth. An input signal $s(t)$ whose spectrum is band-limited to W Hz is passed through the channel.

a. Show that

$$y(t) = s(t) + \frac{1}{2}\alpha[s(t - t_0) + s(t + t_0)]$$

Thus, the channel produces a pair of echoes.

b. Suppose that the received signal $y(t)$ is passed through a filter matched to $s(t)$. Determine the output of the matched filter at $t = kT$, $k = 0, \pm 1, \pm 2, \ldots$, where T is the symbol duration.

c. What is the ISI pattern resulting from the channel if $t_0 = T$?

9.35 A wireline channel of length 1000 km is used to transmit data by means of binary PAM. Regenerative repeaters are spaced 50 km apart along the system. Each segment of the channel has an ideal (constant) frequency response over the frequency band $0 \le f \le 1200$ Hz and an attenuation of 1 dB/km. The channel noise is AWGN.

a. What is the highest bit rate that can be transmitted without ISI?

b. Determine the required $\mathcal{E}_b/N_0$ to achieve a bit error of $P_2 = 10^{-7}$ for each repeater.

c. Determine the transmitted power at each repeater to achieve the desired $\mathcal{E}_b/N_0$, where $N_0 = 4.1 \times 10^{-21}$ W/Hz.

9.36 Prove the relationship in Equation 9.3–13 for the autocorrelation of the noise at the output of the matched filter.

9.37 In the case of PAM with correlated noise, the correlation metrics in the Viterbi algorithm may be expressed in general as (Ungerboeck, 1974)

$$CM(\boldsymbol{I}) = 2\sum_n I_n r_n - \sum_n \sum_m I_n I_m x_{n-m}$$

where $x_n = x(nT)$ is the sampled signal output of the matched filter, $\{I_n\}$ is the data sequence, and $\{r_n\}$ is the received signal sequence at the output of the matched filter. Determine the metric for the duobinary signal.

9.38 Consider the use of a (square-root) raised cosine signal pulse with a roll-off factor of unity for transmission of binary PAM over an ideal band-limited channel that passes the pulse without distortion. Thus, the transmitted signal is

$$v(t) = \sum_{k=-\infty}^{\infty} I_k g_T(t - kT_b)$$

where the signal interval $T_b = \frac{1}{2}T$. Thus, the symbol rate is double of that for no ISI.

a. Determine the ISI values at the output of a matched filter demodulator.

b. Sketch the trellis for the maximum-likelihood sequence detector and label the states.

9.39 A binary antipodal signal is transmitted over a nonideal band-limited channel, which introduces ISI over two adjacent symbols. For an isolated transmitted signal pulse $s(t)$, the (noise-free) output of the demodulator is $\sqrt{\mathcal{E}_b}$ at $t = T$, $\sqrt{\mathcal{E}_b}/4$ at $t = 2T$, and zero for $t = kT$, $k > 2$, where $\mathcal{E}_b$ is the signal energy and T is the signaling interval.

a. Determine the average probability of error, assuming that the two signals are equally probable and the additive noise is white and Gaussian.

b. By plotting the error probability obtained in (a) and that for the case of no ISI, determine the relative difference in SNR of the error probability of 10^{-6}.

9.40 Derive the expression in Equation 9.5–5 for the coefficients in the feedback filter of the DFE.

9.41 Binary PAM is used to transmit information over an unequalized linear filter channel. When $a = 1$ is transmitted, the noise-free output of the demodulator is

$$x_m = \begin{cases} 0.3 & m = 1 \\ 0.9 & m = 0 \\ 0.3 & m = -1 \\ 0 & \text{otherwise} \end{cases}$$

a. Design a three-tap zero-forcing linear equalizer so that the output is

$$q_m = \begin{cases} 1 & m = 0 \\ 0 & m = \pm 1 \end{cases}$$

b. Determine q_m for $m = \pm 2, \pm 3$, by convolving the impulse response of the equalizer with the channel response.

9.42 The transmission of a signal pulse with a raised cosine spectrum through a channel results in the following (noise-free) sampled output from the demodulator:

$$x_k = \begin{cases} -0.5 & k = -2 \\ 0.1 & k = -1 \\ 1 & k = 0 \\ -0.2 & k = 1 \\ 0.05 & k = 2 \\ 0 & \text{otherwise} \end{cases}$$

a. Determine the tap coefficients of a three-tap linear equalizer based on the zero-forcing criterion.

b. For the coefficients determined in (a), determine the output of the equalizer for the case of the isolated pulse. Thus, determine the residual ISI and its span in time.

9.43 A nonideal band-limited channel introduces ISI over three successive symbols. The (noise-free) response of the matched filter demodulator sampled at the sampling time kT is

$$\int_{-\infty}^{\infty} s(t)s(t-kT)\,dt = \begin{cases} \mathcal{E}_b & k = 0 \\ 0.9\mathcal{E}_b & k = \pm 1 \\ 0.1\mathcal{E}_b & k = \pm 2 \\ 0 & \text{otherwise} \end{cases}$$

a. Determine the tap coefficients of a three-tap linear equalizer that equalizes the channel (received signal) response to an equivalent partial-response (duobinary) signal

$$y_k = \begin{cases} \mathcal{E}_b & k = 0, 1 \\ 0 & \text{otherwise} \end{cases}$$

b. Suppose that the linear equalizer in (a) is followed by a Viterbi sequence detector for the partial signal. Give an estimate of the error probability if the additive noise is white and Gaussian, with power spectral density $\frac{1}{2}N_0$ W/Hz.

9.44 Determine the tap weight coefficients of a three-tap zero-forcing equalizer if the ISI spans three symbols and is characterized by the values $x(0) = 1, x(-1) = 0.3, x(1) = 0.2$. Also determine the residual ISI at the output of the equalizer for the optimum tap coefficients.

9.45 In line-of-sight microwave radio transmission, the signal arrives at the receiver via two propagation paths: the direct path and a delayed path that occurs due to signal reflection from surrounding terrain. Suppose that the received signal has the form

$$r(t) = s(t) + \alpha s(t - T) + n(t)$$

where $s(t)$ is the transmitted signal, α is the attenuation ($\alpha < 1$) of the secondary path, and $n(t)$ is AWGN.

a. Determine the output of the demodulator at $t = T$ and $t = 2T$ that employs a filter matched to $s(t)$.

b. Determine the probability of error for a symbol-by-symbol detector if the transmitted signal is binary antipodal and the detector ignores the ISI.

c. What is the error rate performance of a simple (one-tap) DFE that estimates α and removes the ISI? Sketch the detector structure that employs a DFE.

9.46 Repeat Problem 9.41 using the MSE as the criterion for optimizing the tap coefficients. Assume that the noise power spectral density is 0.1 W/Hz.

9.47 In a magnetic recording channel, where the readback pulse resulting from a positive transition in the write current has the form

$$p(t) = \left[1 + \left(\frac{2t}{T_{50}}\right)^2\right]^{-1}$$

a linear equalizer is used to equalize the pulse to a partial response. The parameter T_{50} is defined as the width of the pulse at the 50 percent amplitude level. The bit rate is $1/T_b$ and the ratio of $T_{50}/T_b = \Delta$ is the normalized density of the recording. Suppose the pulse is equalized to the partial-response values

$$x(nT) = \begin{cases} 1 & n = -1, 1 \\ 2 & n = 0 \\ 0 & \text{otherwise} \end{cases}$$

where $x(t)$ represents the equalized pulse shape.

a. Determine the spectrum $X(f)$ of the band-limited equalized pulse.

b. Determine the possible output levels at the detector, assuming that successive transitions can occur at the rate $1/T_b$.

c. Determine the error rate performance of the symbol-by-symbol detector for this signal, assuming that the additive noise is zero-mean Gaussian with variance σ^2.

9.48 Sketch the trellis for the Viterbi detector of the equalized signal in Problem 9.47 and label all the states. Also, determine the minimum Euclidean distance between merging paths.

9.49 Consider the problem of equalizing the discrete-time equivalent channel shown in Figure P9.49. The information sequence $\{I_n\}$ is binary (± 1) and uncorrelated. The additive noise $\{v_n\}$ is white and real-valued, with variance N_0. The received sequence $\{y_n\}$ is processed by a linear three-tap equalizer that is optimized on the basis of the MSE criterion.

a. Determine the optimum coefficients of the equalizer as a function of N_0.

b. Determine the three eigenvalues λ_1, λ_2, and λ_3 of the covariance matrix $\mathbf{\Gamma}$ and the corresponding (normalized to unit length) eigenvectors $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$.

c. Determine the minimum MSE for the three-tap equalizer as a function of N_0.

d. Determine the output SNR for the three-tap equalizer as a function of N_0. How does this compare with the output SNR for the infinite-tap equalizer? For example, evaluate the output SNR for these two equalizers when $N_0 = 0.1$.

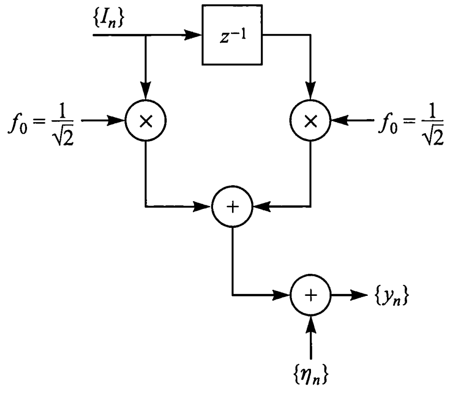


FIGURE P9.49

9.50 Use the orthogonality principle to derive the equations for the coefficients in a decision-feedback equalizer based on the MSE criterion and given by Equations 9.5–3 and 9.5–5.

9.51 Suppose that the discrete-time model for the intersymbol interference is characterized by the tap coefficients $f_0, f_1, \ldots, f_L$. From the equations for the tap coefficients of a decision-feedback equalizer (DFE), show that only L taps are needed in the feedback filter of the DFE. That is, if $\{c_k\}$ are the coefficients of the feedback filter, then $c_k = 0$ for $k \geq L + 1$.

9.52 Consider the channel model shown in Figure P9.52. $\{v_n\}$ is a real-valued white noise sequence with zero-mean and variance N_0. Suppose the channel is to be equalized by a DFE having a two-tap feedforward filter (c_0, c_{-1}) and a one-tap feedback filter (c_1). The $\{c_i\}$ are optimized using the MSE criterion.

a. Determine the optimum coefficients and their approximate values for $N_0 \ll 1$.

b. Determine the exact value of the minimum MSE and a first-order approximation appropriate to the case $N_0 \ll 1$.

c. Determine the exact value of the output SNR for the three-tap equalizer as a function of N_0 and a first-order approximation appropriate to the case $N_0 \ll 1$.

d. Compare the results in (b) and (c) with the performance of the infinite-tap DFE.

e. Evaluate and compare the exact values of the output SNR for the three-tap and infinite-tap DFE in the special cases where $N_0 = 0.1$ and 0.01. Comment on how well the three-tap equalizer performs relative to the infinite-tap equalizer.

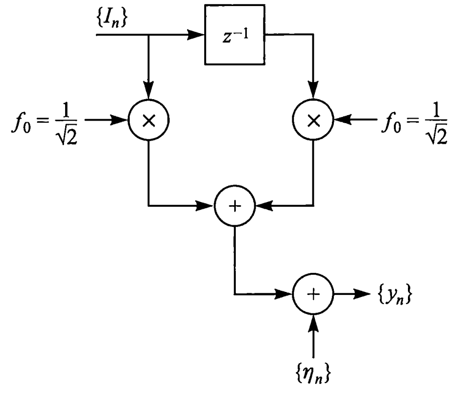


FIGURE P9.52

9.53 A pulse and its (raised cosine) spectral characteristic are shown in Figure P9.53. This pulse is used for transmitting digital information over a band-limited channel at a rate $1/T$ symbols/s.

a. What is the roll-off factor β?

b. What is the pulse rate?

c. The channel distorts the signal pulses. Suppose the sampled values of the filtered received pulse $x(t)$ are as shown in Figure P9.53c. It is obvious that there are five interfering signal components. Give the sequence of +1s and −1s that will cause the largest (destructive or constructive) interference and the corresponding value of the interference (the peak distortion).

d. What is the probability of occurrence of the worst sequence obtained in (c), assuming that all binary digits are equally probable and independent?

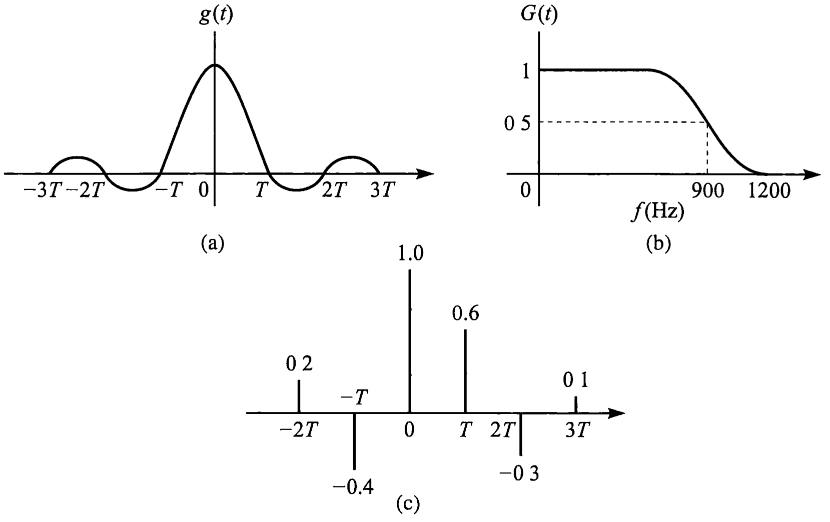


FIGURE P9.53

9.54 A time-dispersive channel having an impulse response $h(t)$ is used to transmit four-phase PSK at a rate $R = 1/T$ symbols/s. The equivalent discrete-time channel is shown in

Figure P9.54. The sequence $\{\eta_k\}$ is a white noise sequence having zero-mean and variance $\sigma^2 = N_0$.

a. What is the sampled autocorrelation function sequence $\{x_k\}$ defined by

$$x_k = \int_{-\infty}^{\infty} h^*(t)h(t + kT)\,dt$$

for this channel?

b. The minimum MSE performance of a linear equalizer and a decision-feedback equalizer having an infinite number of taps depends on the *folded-spectrum of the channel*

$$\frac{1}{T}\sum_{n=-\infty}^{\infty}\left|H\left(\omega + \frac{2\pi n}{T}\right)\right|^2$$

where $H(\omega)$ is the Fourier transform of $h(t)$. Determine the folded spectrum of the channel given above.

c. Use your answer in (b) to express the minimum MSE of a linear equalizer in terms of the folded spectrum of the channel. (You may leave your answer in integral form.)

d. Repeat (c) for an infinite-tap decision-feedback equalizer.

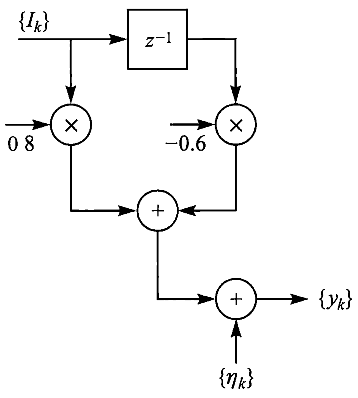


FIGURE P9.54

9.55 Consider a four-level PAM system with possible transmitted levels, 3, 1, −1, and −3. The channel through which the data is transmitted introduces intersymbol interference over two successive symbols. The equivalent discrete-time channel model is shown in Figure P9.55. $\{\eta_k\}$ is a sequence of real-valued independent zero-mean Gaussian noise variables with variance $\sigma^2 = N_0$. The received sequence is

$$\begin{aligned} y_1 &= 0.8I_1 + n_1 \\ y_2 &= 0.8I_2 - 0.6I_1 + n_2 \\ y_3 &= 0.8I_3 - 0.6I_2 + n_3 \\ &\;\vdots \\ y_k &= 0.8I_k - 0.6I_{k-1} + n_k \end{aligned}$$

a. Sketch the tree structure, showing the possible signal sequences for the received signals y_1, y_2, and y_3.

b. Suppose the Viterbi algorithm is used to detect the information sequence. How many probabilities must be computed at each stage of the algorithm?

c. How many surviving sequences are there in the Viterbi algorithm for this channel?

d. Suppose that the received signals are

$$y_1 = 0.5, \qquad y_2 = 2.0, \qquad y_3 = -1.0$$

Determine the surviving sequences through stage y_3 and the corresponding metrics.

e. Give a tight upper bound for the probability of error for four-level PAM transmitted over this channel.

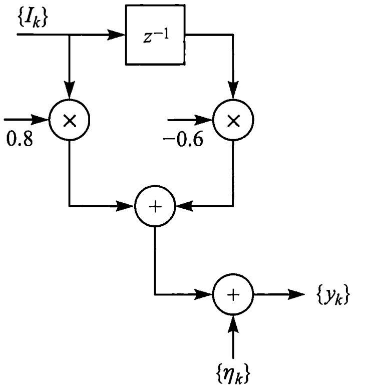


FIGURE P9.55

9.56 A transversal equalizer with K taps has an impulse response

$$e(t) = \sum_{k=0}^{K-1} c_k \delta(t - kT)$$

where T is the delay between adjacent taps, and a transfer function

$$E(z) = \sum_{k=0}^{K-1} c_k z^{-k}$$

The *discrete Fourier transform* (DFT) of the equalizer coefficients $\{c_k\}$ is defined as

$$E_n \equiv E(z)|_{z=e^{j2\pi n/K}} = \sum_{k=0}^{K-1} c_k e^{-j2\pi kn/K}, \qquad n = 0, 1, \ldots, K-1$$

The *inverse DFT* is defined as

$$b_k = \frac{1}{K} \sum_{n=0}^{K-1} E_n e^{j2\pi nk/K}, \qquad k = 0, 1, \ldots, K-1$$

a. Show that $b_k = c_k$, by substituting for E_n in the above expression.

b. From the relations given above, derive an equivalent filter structure having the z transform

$$E(z) = \underbrace{\frac{1 - z^{-K}}{K}}_{E_1(z)} \sum_{n=0}^{K-1} \underbrace{\frac{E_n}{1 - e^{j2\pi n/K} z^{-1}}}_{E_2(z)}$$

c. If $E(z)$ is considered as two separate filters $E_1(z)$ and $E_2(z)$ in cascade, sketch a block diagram for each of the filters, using z^{-1} to denote a unit of delay.

d. In the transversal equalizer, the adjustable parameters are the equalizer coefficients $\{c_k\}$. What are the adjustable parameters of the equivalent equalizer in (b), and how are they related to $\{c_k\}$?

10

Adaptive Equalization

In Chapter 9, we introduced both optimum and suboptimum receivers that compensate for ISI in the transmission of digital information through band-limited, nonideal channels. The optimum receiver employed maximum-likelihood sequence estimation for detecting the information sequence from the samples of the demodulation filter. The suboptimum receivers employed either a linear equalizer or a decision-feedback equalizer.

In the development of the three equalization methods, we implicitly assumed that the channel characteristics, either the impulse response or the frequency response, were known at the receiver. However, in most communication systems that employ equalizers, the channel characteristics are unknown a priori and, in many cases, the channel response is time-variant. In such a case, the equalizers are designed to be adjustable to the channel response and, for time-variant channels, to be adaptive to the time variations in the channel response.

In this chapter, we present algorithms for automatically adjusting the equalizer coefficients to optimize a specified performance index and to adaptively compensate for time variations in the channel characteristics. We also analyze the performance characteristics of the algorithms, including their rate of convergence and their computational complexity.

10.1
ADAPTIVE LINEAR EQUALIZER

In the case of the linear equalizer, recall that we considered two different criteria for determining the values of the equalizer coefficients $\{c_k\}$. One criterion was based on the minimization of the peak distortion at the output of the equalizer, which is defined by Equation 9.4–22. The other criterion was based on the minimization of the mean square error at the output of the equalizer, which is defined by Equation 9.4–42. Below, we describe two algorithms for performing the optimization automatically and adaptively.

10.1–1 The Zero-Forcing Algorithm

In the peak-distortion criterion, the peak distortion $\mathcal{D}(\mathbf{c})$, given by Equation 9.4–22, is minimized by selecting the equalizer coefficients $\{c_k\}$. In general, there is no simple computational algorithm for performing this optimization, except in the special case where the peak distortion at the input to the equalizer, defined as $\mathcal{D}_0$ in Equation 9.4–23, is less than unity. When $\mathcal{D}_0 < 1$, the distortion $\mathcal{D}(\mathbf{c})$ at the output of the equalizer is minimized by forcing the equalizer response $q_n = 0$, for $1 \le |n| \le K$, and $q_0 = 1$. In this case, there is a simple computational algorithm, called the zero-forcing algorithm, that achieves these conditions.

The zero-forcing solution is achieved by forcing the cross correlation between the error sequence $\varepsilon_k = I_k - \hat{I}_k$ and the desired information sequence $\{I_k\}$ to be zero for shifts in the range $0 \le |n| \le K$. The demonstration that this leads to the desired solution is quite simple. We have

$$\begin{aligned} E\left(\varepsilon_k I_{k-j}^*\right) &= E\left[(I_k - \hat{I}_k) I_{k-j}^*\right] \\ &= E\left(I_k I_{k-j}^*\right) - E\left(\hat{I}_k I_{k-j}^*\right), \qquad j = -K, \ldots, K \end{aligned} \tag{10.1–1}$$

We assume that the information symbols are uncorrelated, i.e., $E\left(I_k I_j^*\right) = \delta_{kj}$, and that the information sequence $\{I_k\}$ is uncorrelated with the additive noise sequence $\{\eta_k\}$. For $\hat{I}_k$, we use the expression given in Equation 9.4–41. Then, after taking the expected values in Equation 10.1–1, we obtain

$$E\left(\varepsilon_k I_{k-j}^*\right) = \delta_{j0} - q_j, \qquad j = -K, \ldots, K \tag{10.1–2}$$

Therefore, the conditions

$$E\left(\varepsilon_k I_{k-j}^*\right) = 0, \qquad j = -K, \ldots, K \tag{10.1–3}$$

are fulfilled when $q_0 = 1$ and $q_n = 0, 1 \le |n| \le K$.

When the channel response is unknown, the cross correlations given by Equation 10.1–1 are also unknown. This difficulty can be circumvented by transmitting a known training sequence $\{I_k\}$ to the receiver, which can be used to estimate the cross correlation by substituting time averages for the ensemble averages given in Equation 10.1–1. After the initial training, which will require the transmission of a training sequence of some predetermined length that equals or exceeds the equalizer length, the equalizer coefficients that satisfy Equation 10.1–3 can be determined.

A simple recursive algorithm for adjusting the equalizer coefficients is

$$c_j^{(k+1)} = c_j^{(k)} + \Delta \varepsilon_k I_{k-j}^*, \qquad j = -K, \ldots, -1, 0, 1, \ldots, K \tag{10.1–4}$$

where $c_j^{(k)}$ is the value of the jth coefficient at time $t = kT$, $\varepsilon_k = I_k - \hat{I}_k$ is the error signal at time $t = kT$, and Δ is a scale factor that controls the rate of adjustment, as will be explained later in this section. This is the *zero-forcing algorithm*. The term $\varepsilon_k I_{k-j}^*$ is an estimate of the cross correlation (ensemble average) $E\left(\varepsilon_k I_{k-j}^*\right)$. The averaging operation of the cross correlation is accomplished by means of the recursive first-order difference equation algorithm in Equation 10.1–4, which represents a simple discrete-time integrator.

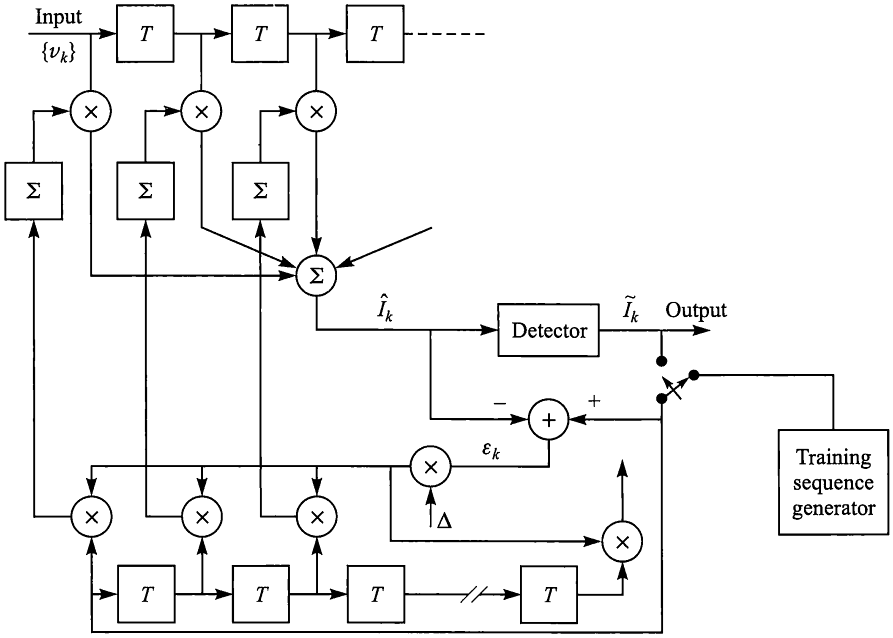


FIGURE 10.1–1
An adaptive zero-forcing equalizer.

Following the training period, after which the equalizer coefficients have converged to their optimum values, the decisions at the output of the detector are generally sufficiently reliable so that they may be used to continue the coefficient adaptation process. This is called a *decision-directed mode* of adaptation. In such a case, the cross correlations in Equation 10.1–4 involve the error signal $\tilde{\varepsilon}_k = \tilde{I}_k - \hat{I}_k$ and the detected output sequence $\tilde{I}_{k-j}$, $j = -K, \ldots, K$. Thus, in the adaptive mode, Equation 10.1–4 becomes

$$c_j^{(k+1)} = c_j^{(k)} + \Delta \tilde{\varepsilon}_k \tilde{I}_{k-j}^* \tag{10.1–5}$$

Figure 10.1–1 illustrates the zero-forcing equalizer in the training mode and the adaptive mode of operation.

The characteristics of the zero-forcing algorithm are similar to those of the least-mean-square (LMS) algorithm, which minimizes the MSE and which is described in detail in the following section.

10.1–2 The LMS Algorithm

In the minimization of the MSE, treated in Section 9.4–2, we found that the optimum equalizer coefficients are determined from the solution of the set of linear equations, expressed in matrix form as

$$\boldsymbol{\Gamma C} = \boldsymbol{\xi} \tag{10.1–6}$$

where $\boldsymbol{\Gamma}$ is the $(2K+1) \times (2K+1)$ covariance matrix of the signal samples $\{v_k\}$, $\boldsymbol{C}$ is the column vector of $(2K+1)$ equalizer coefficients, and $\boldsymbol{\xi}$ is a $(2K+1)$-dimensional column vector of channel filter coefficients. The solution for the optimum equalizer coefficients vector $\boldsymbol{C}_{\text{opt}}$ can be determined by inverting the covariance matrix $\boldsymbol{\Gamma}$, which can be efficiently performed by use of the Levinson–Durbin algorithm (see Levinson (1947) and Durbin (1959)).

Alternatively, an iterative procedure that avoids the direct matrix inversion may be used to compute $\boldsymbol{C}_{\text{opt}}$. Probably the simplest iterative procedure is the method of steepest descent, in which one begins by arbitrarily choosing the vector $\boldsymbol{C}$, say as $\boldsymbol{C}_0$. This initial choice of coefficients corresponds to some point on the quadratic MSE surface in the $(2K+1)$-dimensional space of coefficients. The gradient vector $\boldsymbol{G}_0$, having the $2K+1$ gradient components $\frac{1}{2}\partial J/\partial c_{0k}$, $k = -K, \ldots, -1, 0, 1, \ldots, K$, is then computed at this point on the MSE surface, and each tap weight is changed in the direction opposite to its corresponding gradient component. The change in the jth tap weight is proportional to the size of the jth gradient component. Thus, succeeding values of the coefficient vector $\boldsymbol{C}$ are obtained according to the relation

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k - \Delta \boldsymbol{G}_k, \qquad k = 0, 1, 2, \ldots \tag{10.1–7}$$

where the gradient vector $\boldsymbol{G}_k$ is

$$\boldsymbol{G}_k = \frac{1}{2}\frac{dJ}{d\boldsymbol{C}_k} = \boldsymbol{\Gamma}\boldsymbol{C}_k - \boldsymbol{\xi} = -E\left(\varepsilon_k \boldsymbol{V}_k^*\right) \tag{10.1–8}$$

The vector $\boldsymbol{C}_k$ represents the set of coefficients at the kth iteration, $\varepsilon_k = I_k - \hat{I}_k$ is the error signal at the kth iteration, $\boldsymbol{V}_k$ is the vector of received signal samples that make up the estimate $\hat{I}_k$, i.e., $\boldsymbol{V}_k = [v_{k+K} \cdots v_k \cdots v_{k-K}]^t$, and Δ is a positive number chosen small enough to ensure convergence of the iterative procedure. If the minimum MSE is reached for some $k = k_0$, then $\boldsymbol{G}_k = \mathbf{0}$, so that no further change occurs in the tap weights. In general, $J_{\min}(K)$ cannot be attained for a finite value of k_0 with the steepest-descent method. It can, however, be approached as closely as desired for some finite value of k_0.

The basic difficulty with the method of steepest descent for determining the optimum tap weights is the lack of knowledge of the gradient vector $\boldsymbol{G}_k$, which depends on both the covariance matrix $\boldsymbol{\Gamma}$ and the vector $\boldsymbol{\xi}$ of cross correlations. In turn, these quantities depend on the coefficients $\{f_k\}$ of the equivalent discrete-time channel model and on the covariance of the information sequence and the additive noise, all of which may be unknown at the receiver in general. To overcome the difficulty, estimates of the gradient vector may be used. That is, the algorithm for adjusting the tap weight coefficients may be expressed in the form

$$\hat{\boldsymbol{C}}_{k+1} = \hat{\boldsymbol{C}}_k - \Delta \hat{\boldsymbol{G}}_k \tag{10.1–9}$$

where $\hat{\boldsymbol{G}}_k$ denotes an estimate of the gradient vector $\boldsymbol{G}_k$ and $\hat{\boldsymbol{C}}_k$ denotes the estimate of the vector of coefficients.

From Equation 10.1–8 we note that $\boldsymbol{G}_k$ is the negative of the expected value of the $\varepsilon_k \boldsymbol{V}_k^*$. Consequently, an estimate of $\boldsymbol{G}_k$ is

$$\hat{\boldsymbol{G}}_k = -\varepsilon_k \boldsymbol{V}_k^* \tag{10.1–10}$$

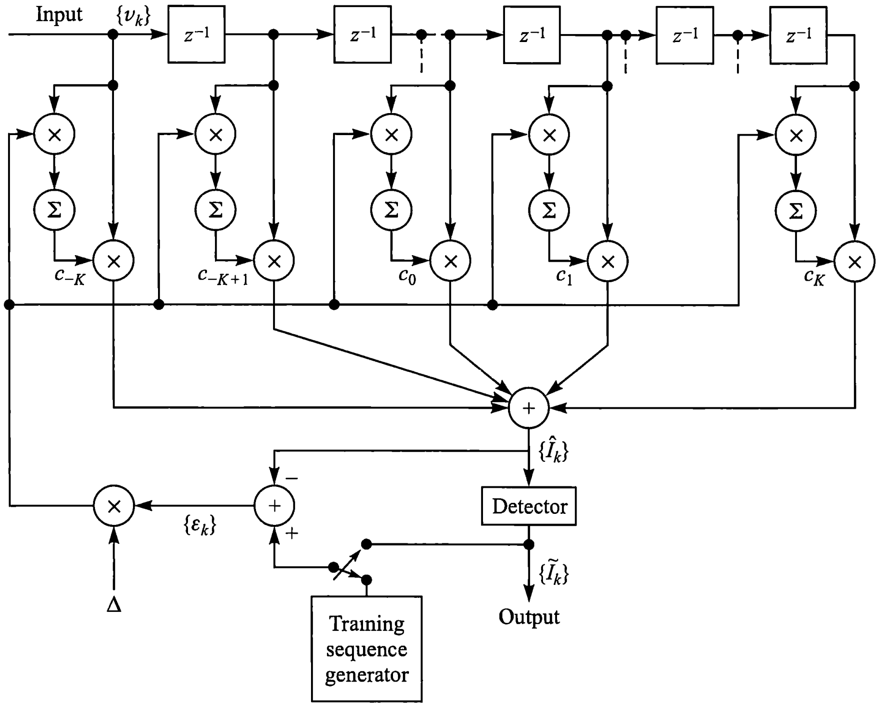


FIGURE 10.1–2
Linear adaptive equalizer based on the MSE criterion.

Since $E(\hat{G}_k) = G_k$, the estimate $\hat{G}_k$ is an unbiased estimate of the true gradient vector $\hat{G}_k$. Incorporation of Equation 10.1–10 into Equation 10.1–9 yields the algorithm

$$\hat{C}_{k+1} = \hat{C}_k + \Delta\varepsilon_k V_k^* \qquad (10.1\text{–}11)$$

This is the basic LMS algorithm for recursively adjusting the tap weight coefficients of the equalizer as described by Widrow (1966). It is illustrated in the equalizer shown in Figure 10.1–2.

The basic algorithm given by Equation 10.1–11 and some of its possible variations have been incorporated into many commercial adaptive equalizers that are used in high-speed modems. Three variations of the basic algorithm are obtained by using only sign information contained in the error signal ε_k and/or in the components of V_k. Hence, the three possible variations are

$$c_{(k+1)j} = c_{kj} + \Delta\, \mathrm{csgn}(\varepsilon_k) v_{k-j}^*, \qquad j = -K, \ldots, -1, 0, 1, \ldots, K \qquad (10.1\text{–}12)$$

$$c_{(k+1)j} = c_{kj} + \Delta\varepsilon_k\, \mathrm{csgn}\big(v_{k-j}^*\big), \qquad j = -K, \ldots, -1, 0, 1, \ldots, K \qquad (10.1\text{–}13)$$

$$c_{(k+1)j} = c_{kj} + \Delta\, \mathrm{csgn}(\varepsilon_k)\, \mathrm{csgn}\big(v_{k-j}^*\big), \qquad j = -K, \ldots, -1, 0, 1, \ldots, K \qquad (10.1\text{–}14)$$

where csgn(x) is defined as

$$\mathrm{csgn}(x) = \begin{cases} 1 + j & [\mathrm{Re}(x) > 0, \mathrm{Im}(x) > 0] \\ 1 - j & [\mathrm{Re}(x) > 0, \mathrm{Im}(x) < 0] \\ -1 + j & [\mathrm{Re}(x) < 0, \mathrm{Im}(x) > 0] \\ -1 - j & [\mathrm{Re}(x) < 0, \mathrm{Im}(x) < 0] \end{cases} \qquad (10.1\text{–}15)$$

(Note that in Equation 10.1–15, $j \equiv \sqrt{-1}$, as distinct from the index j in Equations 10.1–12 to 10.1–14.) Clearly, the algorithm in Equation 10.1–14 is the most easily implemented, but it gives the slowest rate of convergence relative to the others.

Several other variations of the LMS algorithm are obtained by averaging or filtering the gradient vectors over several iterations prior to making adjustments of the equalizer coefficients. For example, the average over N gradient vectors is

$$\bar{\hat{\boldsymbol{G}}}_{mN} = -\frac{1}{N}\sum_{n=0}^{N-1} \varepsilon_{mN+n}\boldsymbol{V}^*_{mN+n} \tag{10.1–16}$$

and the corresponding recursive equation for updating the equalizer coefficients once every N iterations is

$$\hat{\boldsymbol{C}}_{(k+1)N} = \hat{\boldsymbol{C}}_{kN} - \Delta\bar{\hat{\boldsymbol{G}}}_{kN} \tag{10.1–17}$$

In effect, the averaging operation performed in Equation 10.1–16 reduces the noise in the estimate of the gradient vector, as shown by Gardner (1984).

An alternative approach is to filter the noisy gradient vectors by a low-pass filter and use the output of the filter as an estimate of the gradient vector. For example, a simple low-pass filter for the noisy gradients yields as an output

$$\bar{\hat{\boldsymbol{G}}}_k = w\bar{\hat{\boldsymbol{G}}}_{k-1} + (1-w)\hat{\boldsymbol{G}}_k, \qquad \bar{\hat{\boldsymbol{G}}}(0) = \hat{\boldsymbol{G}}(0) \tag{10.1–18}$$

where the choice of $0 \le w < 1$ determines the bandwidth of the low-pass filter. When w is close to unity, the filter bandwidth is small and the effective averaging is performed over many gradient vectors. On the other hand, when w is small, the low-pass filter has a large bandwidth and, hence, it provides little averaging of the gradient vectors. With the filtered gradient vectors given by Equation 10.1–18 in place of $\boldsymbol{G}_k$, we obtain the filtered gradient LMS algorithm given by

$$\hat{\boldsymbol{C}}_{k+1} = \hat{\boldsymbol{C}}_k - \Delta\bar{\hat{\boldsymbol{G}}}_k \tag{10.1–19}$$

In the above discussion, it has been assumed that the receiver has knowledge of the transmitted information sequence in forming the error signal between the desired symbol and its estimate. Such knowledge can be made available during a short training period in which a signal with a known information sequence is transmitted to the receiver for initially adjusting the tap weights. The length of this sequence must be at least as large as the length of the equalizer so that the spectrum of the transmitted signal adequately covers the bandwidth of the channel being equalized.

In practice, the training sequence is often selected to be a periodic pseudorandom sequence, such as a maximum length shift-register sequence whose period N is equal to the length of the equalizer ($N = 2K + 1$). In this case, the gradient is usually averaged over the length of the sequence as indicated in Equation 10.1–16 and the equalizer is adjusted once a period according to Equation 10.1–17. This approach has been called *cyclic equalization*, and has been treated in the papers by Mueller and Spaulding (1975) and Qureshi (1977, 1985). A practical scheme for continuous adjustment of the tap weights may be either a decision-directed mode of operation in which decisions on the information symbols are assumed to be correct and used in place of I_k in forming the error signal ε_k, or one in which a known pseudorandom-probe sequence is inserted in the information-bearing signal either additively or by interleaving in time and the tap weights adjusted by comparing the received probe symbols with the known transmitted probe symbols. In the decision-directed mode of operation, the error signal becomes $\tilde{\varepsilon}_k = \tilde{I}_k - \hat{I}_k$, where $\tilde{I}_k$ is the decision of the receiver based on the estimate $\hat{I}_k$. As long as the receiver is operating at low error rates, an occasional error will have a negligible effect on the convergence of the algorithm.

If the channel response changes, this change is reflected in the coefficients $\{f_k\}$ of the equivalent discrete-time channel model. It is also reflected in the error signal ε_k, since it depends on $\{f_k\}$. Hence, the tap weights will be changed according to Equation 10.1–11 to reflect the change in the channel. A similar change in the tap weights occurs if the statistics of the noise or the information sequence change. Thus, the equalizer is adaptive.

10.1–3 Convergence Properties of the LMS Algorithm

The convergence properties of the LMS algorithm given by Equation 10.1–11 are governed by the step-size parameter Δ. We shall now consider the choice of the parameter Δ to ensure convergence of the steepest-descent algorithm in Equation 10.1–7, which employs the exact value of the gradient.

From Equations 10.1–7 and 10.1–8, we have

$$\begin{aligned} \boldsymbol{C}_{k+1} &= \boldsymbol{C}_k - \Delta \boldsymbol{G}_k \\ &= (\boldsymbol{I} - \Delta\boldsymbol{\Gamma})\boldsymbol{C}_k + \Delta\boldsymbol{\xi} \end{aligned} \tag{10.1–20}$$

where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{\Gamma}$ is the autocorrelation matrix of the received signal, $\boldsymbol{C}_k$ is the $(2K+1)$-dimensional vector of equalizer tap gains, and $\boldsymbol{\xi}$ is the vector of cross correlations given by Equation 9.4–45. The recursive relation in Equation 10.1–20 can be represented as a closed-loop control system as shown in Figure 10.1–3. Unfortunately, the set of $2K+1$ first-order difference equations in Equation 10.1–20 are coupled through the autocorrelation matrix $\boldsymbol{\Gamma}$. In order to solve these equations and, thus, establish the convergence properties of the recursive algorithm, it is mathematically convenient to decouple the equations by performing a linear transformation. The appropriate transformation is obtained by noting that the matrix $\boldsymbol{\Gamma}$ is Hermitian and, hence, can be represented as

$$\boldsymbol{\Gamma} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^H \tag{10.1–21}$$

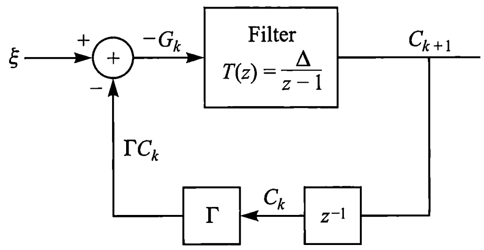


FIGURE 10.1–3
Closed-loop control system representation of the recursive relation in Equation 10.1–20.

where $\boldsymbol{U}$ is the normalized modal matrix of $\boldsymbol{\Gamma}$ and $\boldsymbol{\Lambda}$ is a diagonal matrix with diagonal elements equal to the eigenvalues of $\boldsymbol{\Gamma}$ (see Appendix A).

When Equation 10.1–21 is substituted into Equation 10.1–20 and if we define the transformed (orthogonalized) vectors $\boldsymbol{C}_k^o = \boldsymbol{U}^H \boldsymbol{C}_k$ and $\boldsymbol{\xi}^o = \boldsymbol{U}^H \boldsymbol{\xi}$, we obtain

$$\boldsymbol{C}_{k+1}^o = (\boldsymbol{I} - \Delta\boldsymbol{\Lambda})\boldsymbol{C}_k^o + \Delta\boldsymbol{\xi}^o \qquad (10.1\text{–}22)$$

This set of first-order difference equations is now decoupled. Their convergence is determined from the homogeneous equation

$$\boldsymbol{C}_{k+1}^o = (\boldsymbol{I} - \Delta\boldsymbol{\Lambda})\boldsymbol{C}_k^o \qquad (10.1\text{–}23)$$

We see that the recursive relation will converge provided that all the poles lie inside the unit circle, i.e.,

$$|1 - \Delta\lambda_k| < 1, \qquad k = -K, \ldots, -1, 0, 1, \ldots, K \qquad (10.1\text{–}24)$$

where $\{\lambda_k\}$ is the set of $2K + 1$ (possibly nondistinct) eigenvalues of $\boldsymbol{\Gamma}$. Since $\boldsymbol{\Gamma}$ is an autocorrelation matrix, it is positive-definite and, hence, $\lambda_k > 0$ for all k. Consequently convergence of the recursive relation in Equation 10.1–22 is ensured if Δ satisfies the inequality

$$0 < \Delta < \frac{2}{\lambda_{\max}} \qquad (10.1\text{–}25)$$

where $\lambda_{\max}$ is the largest eigenvalue of $\boldsymbol{\Gamma}$.

Since the largest eigenvalue of a positive-definite matrix is less than the sum of all the eigenvalues of the matrix and, furthermore, since the sum of the eigenvalues of a matrix is equal to its trace, we have the following simple upper bound on $\lambda_{\max}$:

$$\begin{aligned} \lambda_{\max} &< \sum_{k=-K}^{K} \lambda_k = \operatorname{tr} \boldsymbol{\Gamma} = (2K + 1)\Gamma_{kk} \\ &= (2K + 1)(x_0 + N_0) \end{aligned} \qquad (10.1\text{–}26)$$

From Equations 10.1–23 and 10.1–24 we observe that rapid convergence occurs when $|1 - \Delta\lambda_k|$ is small, i.e., when the pole positions are far from the unit circle. But we cannot achieve this desirable condition and still satisfy Equation 10.1–25 if there is a large difference between the largest and smallest eigenvalues of $\boldsymbol{\Gamma}$. In other words, even if we select Δ to be near the upper bound given in Equation 10.1–25, the convergence rate of the recursive MSE algorithm is determined by the smallest eigenvalue $\lambda_{\min}$. Consequently, the ratio $\lambda_{\max}/\lambda_{\min}$ ultimately determines the convergence rate. If $\lambda_{\max}/\lambda_{\min}$ is small, Δ can be selected so as to achieve rapid convergence. However, if the ratio $\lambda_{\max}/\lambda_{\min}$ is large, as is the case when the channel frequency response has deep spectral nulls, the convergence rate of the algorithm will be slow.

10.1–4 Excess MSE due to Noisy Gradient Estimates

The recursive algorithm in Equation 10.1–11 for adjusting the coefficients of the linear equalizer employs unbiased noisy estimates of the gradient vector. The noise in these estimates causes random fluctuations in the coefficients about their optimal values and, thus, leads to an increase in the MSE at the output of the equalizer. That is, the final MSE is $J_{\min} + J_\Delta$, where J_Δ is the variance of the measurement noise. The term J_Δ due to the estimation noise has been termed *excess mean square error* by Widrow (1966).

The total MSE at the output of the equalizer for any set of coefficients $\boldsymbol{C}$ can be expressed as

$$J = J_{\min} + (\boldsymbol{C} - \boldsymbol{C}_{\text{opt}})^H \boldsymbol{\Gamma} (\boldsymbol{C} - \boldsymbol{C}_{\text{opt}}) \tag{10.1–27}$$

where $\boldsymbol{C}_{\text{opt}}$ represents the optimum coefficients, which satisfy Equation 10.1–6. This expression for the MSE can be simplified by performing the linear orthogonal transformation used above to establish convergence. The result of this transformation applied to Equation 10.1–27 is

$$J = J_{\min} + \sum_{k=-K}^{K} \lambda_k E|c_k^o - c_{k\,\text{opt}}^o|^2 \tag{10.1–28}$$

where the $\{c_k^o\}$ are the set of transformed equalizer coefficients. The excess MSE is the expected value of the second term in Equation 10.1–28, i.e.,

$$J_\Delta = \sum_{k=-K}^{K} \lambda_k E|c_k^o - c_{k\,\text{opt}}^o|^2 \tag{10.1–29}$$

It has been shown by Widrow (1970) that the excess MSE is

$$J_\Delta = \Delta^2 J_{\min} \sum_{k=-K}^{K} \frac{\lambda_k^2}{1 - (1 - \Delta\lambda_k)^2} \tag{10.1–30}$$

The expression in Equation 10.1–30 can be simplified when Δ is selected such that $\Delta\lambda_k \ll 1$ for all k. Then

$$\begin{aligned} J_\Delta &\approx \tfrac{1}{2}\Delta J_{\min} \sum_{k=-K}^{K} \lambda_k \\ &\approx \tfrac{1}{2}\Delta J_{\min} \operatorname{tr} \boldsymbol{\Gamma} \\ &\approx \tfrac{1}{2}\Delta (2K+1) J_{\min}(x_0 + N_0) \end{aligned} \tag{10.1–31}$$

Note that $x_0 + N_0$ represents the received signal plus noise power.

It is desirable to have $J_\Delta < J_{\min}$. That is, Δ should be selected such that

$$\frac{J_\Delta}{J_{\min}} \approx \tfrac{1}{2}\Delta(2K+1)(x_0 + N_0) < 1$$

or, equivalently,

$$\Delta < \frac{2}{(2K+1)(x_0 + N_0)} \tag{10.1–32}$$

For example, if Δ is selected as

$$\Delta = \frac{0.2}{(2K+1)(x_0 + N_0)} \tag{10.1–33}$$

the degradation in the output SNR of the equalizer due to the excess MSE is less than 1 dB.

The analysis given above on the excess mean square error is based on the assumption that the mean value of the equalizer coefficients has converged to the optimum value $\boldsymbol{C}_{\text{opt}}$. Under this condition, the step size Δ should satisfy the bound in Equation 10.1–32. On the other hand, we have determined that convergence of the mean coefficient vector requires that $\Delta < 2/\lambda_{\max}$. While a choice of Δ near the upper bound $2/\lambda_{\max}$ may lead to initial convergence of the deterministic (known) steepest-descent gradient algorithm, such a large value of Δ will usually result in instability of the LMS stochastic gradient algorithm.

The initial convergence or transient behavior of the LMS algorithm has been investigated by several researchers. Their results clearly indicate that the step size must be reduced in direct proportion to the length of the equalizer as specified by Equation 10.1–32. Hence, the upper bound given by Equation 10.1–32 is also necessary to ensure the initial convergence of the LMS algorithm. The papers by Gitlin and Weinstein (1979) and Ungerboeck (1972) contain analyses of the transient behavior and the convergence properties of the LMS algorithm.

The following example serves to reinforce the important points made above regarding the initial convergence of the LMS algorithm.

EXAMPLE 10.1-1. The LMS algorithm was used to adaptively equalize a communication channel for which the autocorrelation matrix $\boldsymbol{\Gamma}$ has an eigenvalue spread of $\lambda_{\max}/\lambda_{\min} = 11$. The number of taps selected for the equalizer was $2K + 1 = 11$. The input signal plus noise power $x_0 + N_0$ was normalized to unity. Hence, the upper bound on Δ given by Equation 10.1–32 is 0.18. Figure 10.1–4 illustrates the initial convergence characteristics of the LMS algorithm for $\Delta = 0.045, 0.09$, and 0.115, by averaging the (estimated) MSE in 200 simulations. We observe that by selecting $\Delta = 0.09$ (one-half of the upper bound) we obtain relatively fast initial convergence. If we divide Δ by a factor of 2 to $\Delta = 0.045$, the convergence rate is reduced but the excess mean square error is also reduced, so that the LMS algorithm performs better in steady state (in a time-invariant signal environment). Finally, we note that a choice of $\Delta = 0.115$, which

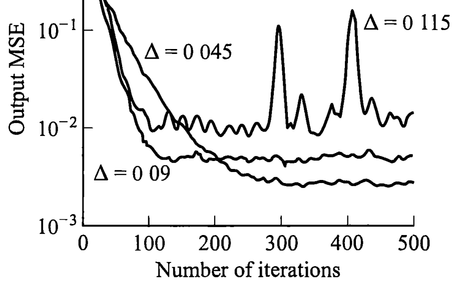


FIGURE 10.1–4
Initial convergence characteristics of the LMS algorithm with different step sizes. (*From* Digital Signal Processing, *by J. G. Proakis and D. G. Manolakis, 1995, Prentice Hall Company. Reprinted with permission of the publisher.*)

is still far below the upper bound, causes large undesirable fluctuations in the output MSE of the algorithm.

In a digital implementation of the LMS algorithm, the choice of the step-size parameter becomes even more critical. In an attempt to reduce the excess mean square error, it is possible to reduce the step-size parameter to the point where the total mean square error actually increases. This condition occurs when the estimated gradient components of the vector $\varepsilon_k \boldsymbol{V}_k^*$ after multiplication by the small step-size parameter Δ are smaller than one-half of the least significant bit in the fixed-point representation of the equalizer coefficients. In such a case, adaptation ceases. Consequently, it is important for the step size to be large enough to bring the equalizer coefficients in the vicinity of $\boldsymbol{C}_{\mathrm{opt}}$. If it is desired to decrease the step size significantly, it is necessary to increase the precision in the equalizer coefficients. Typically, 16 bits of precision may be used for the coefficients, with about 10–12 of the most significant bits used for arithmetic operations in the equalization of the data. The remaining least significant bits are required to provide the necessary precision for the adaptation process. Thus, the scaled estimated gradient components $\Delta \varepsilon \boldsymbol{V}_k^*$ usually affect only the least-significant bits in any one iteration. In effect, the added precision also allows for the noise to be averaged out, since many incremental changes in the least-significant bits are required before any change occurs in the upper more significant bits used in arithmetic operations for equalizing the data. For an analysis of roundoff errors in a digital implementation of the LMS algorithm, the reader is referred to the papers by Gitlin and Weinstein (1979), Gitlin et al. (1982), and Caraiscos and Liu (1984).

As a final point, we should indicate that the LMS algorithm is appropriate for tracking slowly time invariant signal statistics. In such a case, the minimum MSE and the optimum coefficient vector will be time-variant. In other words, $J_{\min}(n)$ is a function of time and the $2(K + 1)$-dimensional error surface is moving with the time index n. The LMS algorithm attempts to follow the moving minimum $J_{\min}(n)$ in the $(2K + 1)$-dimensional space, but it is always lagging behind due to its use of (estimated) gradient vectors. As a consequence, the LMS algorithm incurs another form of error, called the *lag error*, whose mean square value decreases with an increase in the step size Δ. The total MSE error can now be expressed as

$$J_{\mathrm{total}} = J_{\min}(n) + J_{\Delta} + J_l \tag{10.1–34}$$

where J_l denotes the mean square error due to the lag.

In any given nonstationary adaptive equalization problem, if we plot the errors J_{Δ} and J_l as a function of Δ, we expect these errors to behave as illustrated in Figure 10.1–5. We observe that J_{Δ} increases with an increase in Δ while J_l decreases with an increase in Δ. The total error will exhibit a minimum, which will determine the optimum choice of the step-size parameter.

When the statistical time variations of the signal occur rapidly, the lag error will dominate the performance of the adaptive equalizer. In such a case, $J_l \gg J_{\min} + J_{\Delta}$, even when the largest possible value of Δ is used. When this condition occurs, the LMS algorithm is inappropriate for the application and one must rely on the more complex recursive least-squares algorithms described in Section 10.4 to obtain faster convergence.

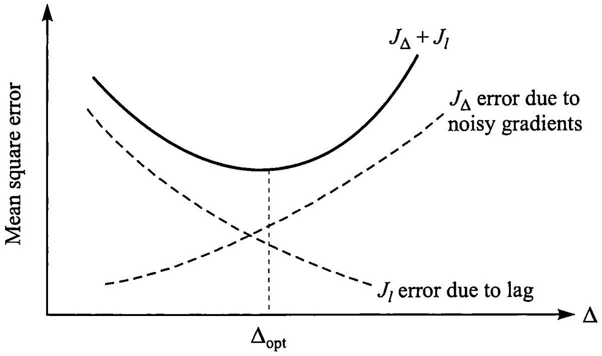


FIGURE 10.1–5
Excess mean square error J_Δ and lag error J_l as a function of the step size. (*From* Digital Signal Processing, *by J. G. Proakis and D. G. Manolakis, 1995, Prentice Hall Company. Reprinted with permission of the publisher.*)

10.1–5 Accelerating the Initial Convergence Rate in the LMS Algorithm

As we have observed, the initial convergence rate of the LMS algorithm for any given channel characteristic is controlled by the step-size parameter Δ. The initial convergence rate is strongly influenced by the channel spectral characteristics, which are related to the eigenvalues $\{\lambda_n\}$ of the received signal covariance matrix. If the channel amplitude and phase distortions are small, the eigenvalue ratio $\lambda_{\max}/\lambda_{\min}$ is close to unity and, hence, the equalizer converges to its optimum tap coefficients relatively fast. On the other hand, if the channel exhibits poor spectral characteristics, such as relatively large attenuation in a part of its spectrum, the eigenvalue ratio $\lambda_{\max}/\lambda_{\min} \gg 1$ and, hence, the convergence rate of the LMS algorithm will be slow.

A considerable effort has been spent by researchers on methods to accelerate the initial convergence of the LMS algorithm. A simple remedy is to begin with a large step size, say Δ_0, and reduce the step size as the tap coefficients converge to their optimum values. In other words, we use a sequence of step sizes, $\Delta_0 > \Delta_1 > \Delta_2 > \cdots > \Delta_m \equiv \Delta$, where Δ is the final step size to be used in steady-state operation of the LMS algorithm.

An alternative method for accelerating initial convergence has been proposed and investigated by Chang (1971) and Qureshi (1977). This method is based on introducing additional parameters in the LMS algorithm by replacing the step size with a weighting matrix $\boldsymbol{W}$. In such a case, the LMS algorithm is generalized to the form:

$$\begin{aligned}\hat{\boldsymbol{C}}_{k+1} &= \hat{\boldsymbol{C}}_k - \boldsymbol{W}\hat{\boldsymbol{G}}_k \\ &= \hat{\boldsymbol{C}}_k + \boldsymbol{W}(\boldsymbol{\Gamma}\hat{\boldsymbol{C}} - \boldsymbol{\xi}) \\ &= \hat{\boldsymbol{C}}_k + \boldsymbol{W}e_k\boldsymbol{V}_k^*\end{aligned} \tag{10.1–35}$$

where $\boldsymbol{W}$ is the weighting matrix. Ideally, $\boldsymbol{W} = \boldsymbol{\Gamma}^{-1}$, or if $\boldsymbol{\Gamma}$ is estimated, then $\boldsymbol{W}$ can be set equal to the inverse of the estimate.

When the training sequence for the equalizer is periodic with period N, the covariance matrix $\boldsymbol{\Gamma}$ is Toeplitz and circulant and its inverse is circulant. In this case, the multiplication by the weighting matrix $\boldsymbol{W}$ can be simplified considerably by the implementation of a single finite duration impulse response (FIR) filter with weights equal to the first row of $\boldsymbol{W}$, as indicated by Qureshi (1977). That is, the fast update algorithm that is equivalent to multiplying the gradient vector $\hat{\boldsymbol{G}}_k$ by $\boldsymbol{W}$ is simply implemented as shown in Figure 10.1–6, by inserting the FIR filter with N coefficients

$w_0, w_1, \ldots, w_{N-1}$ in the path of the periodic input sequence before it is used for tap coefficient adjustment.

Qureshi (1977) described a method for estimating the weights from the received signal. The basic steps are as follows:

1. Collect one period (N symbols) of received data $v_0, v_1, \ldots, v_{N-1}$ in the equalizer delay line.
2. Compute the N-point discrete Fourier transform (DFT) of $\{v_n\}$ denoted as $\{R_n\}$.
3. Compute the discrete power spectrum $|R_n|^2$. If we neglect the noise, $|R_n|^2$ corresponds to N times the eigenvalues of the circulant covariance matrix of the signal at the input to the equalizer. Then, add N times the estimate of the noise variance σ^2 to $|R_n|^2$.
4. Compute the inverse DFT of the sequence $1/\big(|R_n|^2 + N\hat{\sigma}^2\big)$, $n = 0, 1, \ldots, N-1$. This yields the sequence $\{w_n\}$ of filter coefficients for the filter shown in Figure 10.1–6.
5. The algorithm for adjusting the equalizer tap coefficient now becomes

$$c_j^{(k+1)} = c_j^{(k)} - e_j \sum_{m=0}^{N-1} w_k v_{k-j-m}^*, \qquad j = 0, 1, \ldots, N-1 \tag{10.1–36}$$

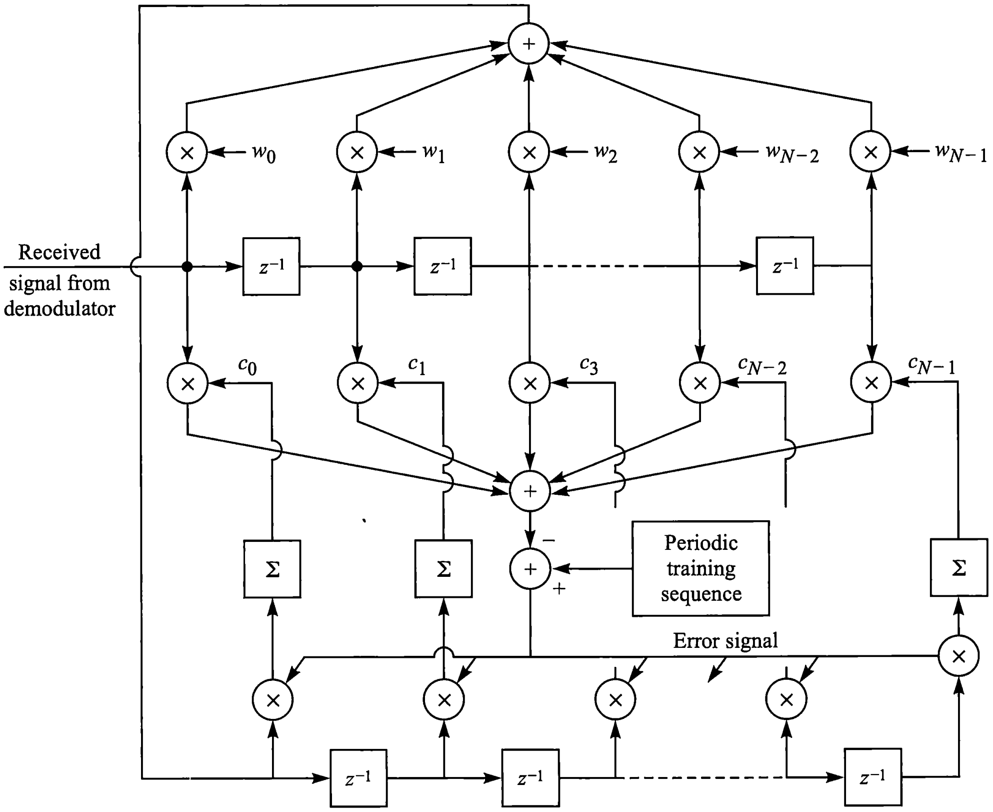

+
w_0
w_1
w_2
w_{N-2}
w_{N-1}
Received signal from demodulator
z^{-1}
z^{-1}
z^{-1}
c_0
c_1
c_3
c_{N-2}
c_{N-1}
+
−
+
+
Σ
Σ
Σ
Periodic training sequence
Error signal
z^{-1}
z^{-1}
z^{-1}

FIGURE 10.1–6
Fast start-up technique for an adaptive equalizer.

10.1–6 Adaptive Fractionally Spaced Equalizer—The Tap Leakage Algorithm

As described in Section 9.4–4, an FSE is preferable to a symbol rate equalizer (SRE) when the channel characteristics are unknown at the receiver. In such a case, the FSE combines the operations of matched filtering and equalization of intersymbol interference into a single filter. By processing samples at the Nyquist rate, the FSE adapts its coefficients to compensate for any timing phase within a symbol. Thus, its performance is insensitive to the sampling time within a symbol interval, as discussed previously. Consequently, from a performance viewpoint, the FSE is equivalent to a matched filter followed by a symbol rate sampler, and followed by an SRE.

The LMS algorithm and any of its variants can be used to adjust the coefficients of the FSE adaptively. Suitable training signals for initial adjustment may take the form of an aperiodic pseudorandom sequence or a periodic pseudorandom sequence, where the period is equal to the time span of the equalizer, i.e., a sequence of period P is used to train an FSE with PN/M coefficients, where the tap spacing is MT/N. In the case of a periodic sequence for training, the update of each of the coefficients may be performed periodically, once in every period of the sequence based on the average gradient LMS algorithm given by Equations 10.1–16 and 10.1–17.

In a digital implementation of the LMS algorithm for an FSE, some care must be exercised in selecting the step-size parameter Δ. It has been shown by Gitlin and Weinstein (1981) and further described by Qureshi (1985) that in an FSE, a fraction $(N - M)/N$ of the eigenvalues of the received signal covariance matrix are very small. These small eigenvalues and their corresponding eigenvectors are related to the spectral characteristics of the noise in the frequency band $(1 + \beta)/2T \leq |f| \leq 1/T$. As a consequence, the output MSE becomes insensitive to deviations in the coefficient values corresponding to these eigenvalues. In such cases, errors due to finite precision arithmetic accumulate along the eigenvectors (frequency band) corresponding to the small eigenvalues and eventually cause overflows in the coefficient values, without significantly affecting the overall MSE.

A solution to this problem has been given in the paper by Gitlin et al. (1982). Instead of minimizing the MSE given by Equation 9.4–42, we minimize the performance index

$$J = J_{\text{MSE}} + \mu \sum_{i=-K}^{K} |c_i|^2 \tag{10.1–37}$$

where J_{MSE} is the conventional MSE and μ is a small positive constant. Thus, the ill-conditioning of the received signal covariance matrix is avoided. The minimization of J leads to the following "modified LMS" algorithm (see Problem 10.5).

$$\boldsymbol{C}_{k+1} = (1 - \Delta\mu)\boldsymbol{C}_k + \Delta\varepsilon_k \boldsymbol{V}_k^* \tag{10.1–38}$$

This algorithm is called the *tap-leakage algorithm.*

In adapting the tap coefficients of an FSE, the tap adjustments, as described above, are made periodically either at the symbol rate or slower when a periodic training sequence is transmitted. However, the samples at the input to the FSE occur at a faster rate. For example, if we consider a $T/2$ FSE, there are two samples per information symbol. An interesting question is whether or not it is possible to increase the initial convergence rate of an FSE by adapting its coefficients at the sampling rate. If the tap adjustments are performed at the sampling rate, one must generate additional desired signal values corresponding to sample values that fall between values of the desired symbols. That is, one must design a filter that performs intersymbol interpolation in order to generate the intermediate desired sample sequence. This problem has been considered by Gitlin and Weinstein (1981), Cioffi and Kailath (1984), and Ling (1989). The results given in the paper by Ling provide an answer to the question.

First we note that the initial convergence of the LMS algorithm depends on the number of nontrivial eigenvalues of the autocorrelation matrix of the received signal. This number is equal to the number of independent parameters that are to be optimized. For example, an SRE that has K taps and spans a time interval of KT seconds has K independent parameters to be optimized. In contrast, a $T/2$ complex-valued FSE that spans the same time interval has $2K$ tap coefficients, but its autocorrelation matrix has K nontrivial (and K trivial) eigenvalues and, thus, it has K independent parameters to be optimized. Consequently, the complex-valued $T/2$ FSE that is adapted at the symbol rate has the same convergence rate as the SRE. Now, if the complex-valued FSE employs interpolation to update its coefficients at all time instants $nT/2$, the number of independent parameters to be optimized is $2K$. In this case, there are two autocorrelation matrices, one corresponding to samples at $nT/2$, and the other corresponding to samples at $(nT+1)/2$, and each matrix has K nontrivial eigenvalues. That is, the $T/2$ FSE that employs interpolation adjusts one set of K parameters in one update and the second set of K parameters in the next update. Therefore, the convergence rate of the interpolated FSE will be approximately the same as the convergence rate of the symbol-updated FSE.

In the case of a phase-splitting FSE (PS-FSE), which is implemented at bandpass, with a time span of KT seconds and tap spacing T/N, where $N > 2$, e.g., $N = 3$ or 4, there are KN parameters to be optimized. In this case, Ling (1989) showed that the convergence rate of the PS-FSE was approximately a factor of 2 slower than the convergence rate of the conventional complex-valued FSE, when the PS-FSE is adjusted at the symbol rate. By employing ideal intersymbol interpolation, the convergence rate of the PS-FSE is increased by approximately a factor of 2 compared to symbol rate adjustment of the PS-FSE. Thus, the PS-FSE with intersymbol interpolation achieves the same convergence rate as the conventional complex-valued FSE that is adjusted at the symbol rate.

10.1–7 An Adaptive Channel Estimator for ML Sequence Detection

The ML sequence detection criterion implemented via the Viterbi algorithm as embodied in the metric computation given by Equation 9.3–23 requires knowledge of the equivalent discrete-time channel coefficients $\{f_k\}$. To accommodate a channel that is unknown or slowly time varying, one may include a channel estimator connected in parallel with the detection algorithm, as shown in Figure 10.1–7. The channel estimator, which is shown in Figure 10.1–8, is identical in structure to the linear transversal equalizer discussed previously in Section 10.1. In fact, the channel estimator is a replica of the equivalent discrete-time channel filter that models the intersymbol

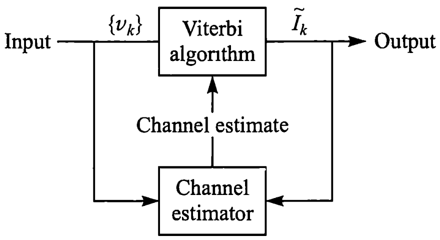


FIGURE 10.1–7
Block diagram of method for estimating the channel characteristics for the Viterbi algorithm.

interference. The estimated tap coefficients, denoted by $\{\hat{f}_k\}$, are adjusted recursively to minimize the MSE between the actual received sequence and the output of the estimator. For example, the LMS steepest-descent algorithm in a decision-directed mode of operation is

$$\hat{\boldsymbol{f}}_{k+1} = \hat{\boldsymbol{f}}_k + \Delta \varepsilon_k \tilde{\boldsymbol{I}}_k^* \tag{10.1–39}$$

where $\hat{\boldsymbol{f}}_k$ is the vector of tap gain coefficients at the kth iteration, Δ is the step size, $\varepsilon_k = v_k - \hat{v}_k$ is the error signal, and $\tilde{\boldsymbol{I}}_k$ denotes the vector of detected information symbols in the channel estimator at the kth iteration.

We now show that when the MSE between v_k and $\hat{v}_k$ is minimized, the resulting values of the tap gain coefficients of the channel estimator are the values of the discrete-time channel model. For mathematical tractability, we assume that the detected information sequence $\{\tilde{I}_k\}$ is correct, i.e., $\{\tilde{I}_k\}$ is identical to the transmitted sequence $\{I_k\}$. This is a reasonable assumption when the system is operating at a low probability of error. Thus, the MSE between the received signal v_k and the estimate $\hat{v}_k$ is

$$J(\hat{\boldsymbol{f}}) = E\left(\left|v_k - \sum_{j=0}^{N-1} \hat{f}_j I_{k-j}\right|^2\right) \tag{10.1–40}$$

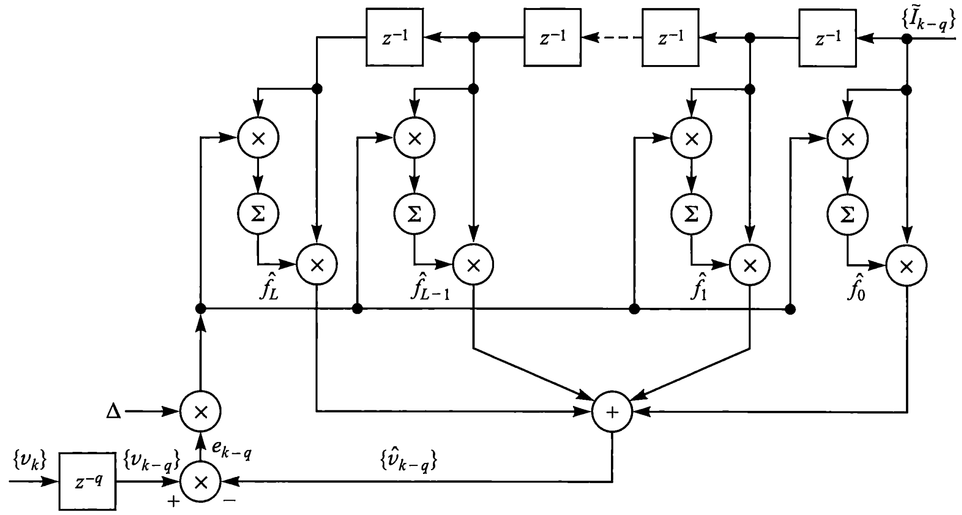


FIGURE 10.1–8
Adaptive transversal filter for estimating the channel dispersion.

The tap coefficients $\{\hat{f}_k\}$ that minimize $J(\hat{\boldsymbol{f}})$ in Equation 10.1–40 satisfy the set of N linear equations

$$\sum_{j=0}^{N-1} \hat{f}_j R_{kj} = d_k, \qquad k = 0, 1, \ldots, N-1 \tag{10.1–41}$$

where

$$R_{kj} = E\left(I_k I_j^*\right), \qquad d_k = \sum_{j=0}^{N-1} f_j R_{kj} \tag{10.1–42}$$

From Equations 10.1–41 and 10.1–42, we conclude that, as long as the information sequence $\{I_k\}$ is uncorrelated, the optimum coefficients are exactly equal to the respective values of the equivalent discrete-time channel. It is also apparent that when the number of taps N in the channel estimator is greater than or equal to $L+1$, the optimum tap gain coefficients $\{\hat{f}_k\}$ are equal to the respective values of the $\{f_k\}$, even when the information sequence is correlated. Subject to the above conditions, the minimum MSE is simply equal to the noise variance N_0.

In the above discussion, the estimated information sequence at the output of the Viterbi algorithm or the probabilistic symbol-by-symbol algorithm was used in making adjustments of the channel estimator. For start-up operation, one may send a short training sequence to perform the initial adjustment of the tap coefficients, as is usually done in the case of the linear transversal equalizer. In an adaptive mode of operation, the receiver simply uses its own decisions to form an error signal.

10.2
ADAPTIVE DECISION-FEEDBACK EQUALIZER

As in the case of the linear adaptive equalizer, the coefficients of the feedforward filter and the feedback filter in a decision-feedback equalizer (DFE) may be adjusted recursively, instead of inverting a matrix as implied by Equation 9.5–3. Based on the minimization of the MSE at the output of the DFE, the steepest-descent algorithm takes the form

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k + \Delta E\left(\varepsilon_k \boldsymbol{V}_k^*\right) \tag{10.2–1}$$

where $\boldsymbol{C}_k$ is the vector of equalizer coefficients in the kth signal interval, $E\left(\varepsilon_k \boldsymbol{V}_k^*\right)$ is the cross correlation of the error signal $\varepsilon_k = I_k - \hat{I}_k$ with $\boldsymbol{V}_k = [v_{k+K_1} \cdots v_k \, I_{k-1} \cdots I_{k-K_2}]^t$, representing the signal values in the feedforward and feedback filters at time $t = kT$. The MSE is minimized when the cross-correlation vector $E\left(\varepsilon_k \boldsymbol{V}_k^*\right) = 0$ as $k \to \infty$.

Since the exact cross-correlation vector is unknown at any time instant, we use as an estimate the vector $\varepsilon_k \boldsymbol{V}_k^*$ and average out the noise in the estimate through the recursive equation

$$\hat{\boldsymbol{C}}_{k+1} = \hat{\boldsymbol{C}}_k + \Delta \varepsilon_k \boldsymbol{V}_k^* \tag{10.2–2}$$

This is the LMS algorithm for the DFE.

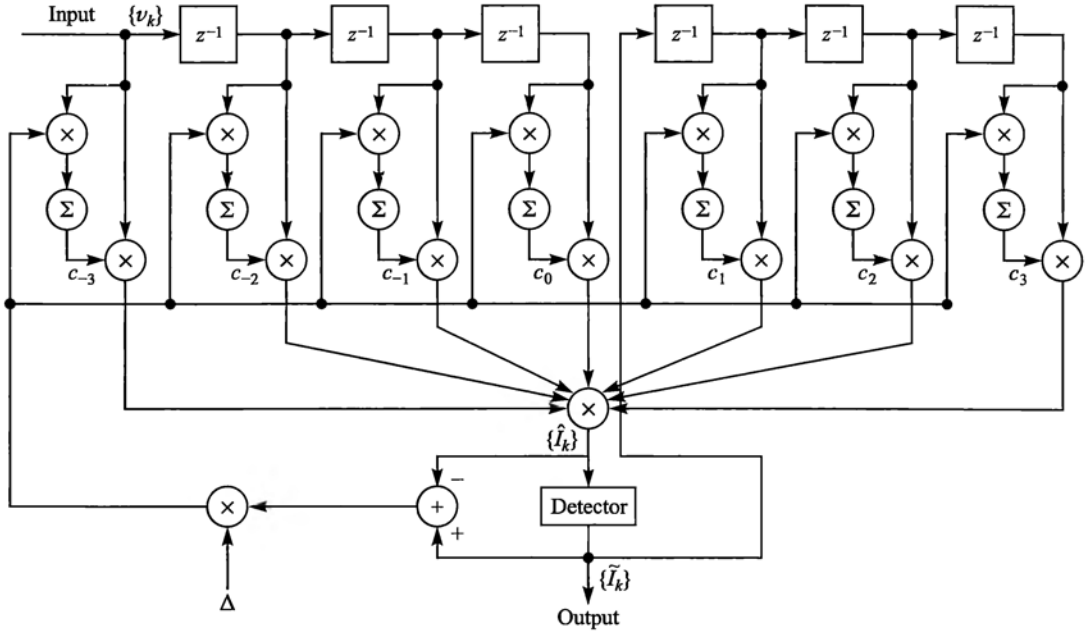


FIGURE 10.2–1
Decision-feedback equalizer.

As in the case of a linear equalizer, we may use a training sequence to adjust the coefficients of the DFE initially. Upon convergence to the (near-) optimum coefficients (minimum MSE), we may switch to a decision-directed mode where the decisions at the output of the detector are used in forming the error signal ε_k and fed to the feedback filter. This is the adaptive mode of the DFE, which is illustrated in Figure 10.2–1. In this case, the recursive equation for adjusting the equalizer coefficient is

$$\tilde{\boldsymbol{C}}_{k+1} = \tilde{\boldsymbol{C}}_k + \Delta \tilde{\varepsilon}_k \boldsymbol{V}_k^* \tag{10.2–3}$$

where $\tilde{\varepsilon}_k = \tilde{I}_k - \hat{I}_k$ and $\boldsymbol{V}_k = [v_{k+K_1} \cdots v_k \tilde{I}_{k-1} \cdots \tilde{I}_{k-K_2}]^t$.

The performance characteristics of the LMS algorithm for the DFE are basically the same as the development given in Sections 10.1–3 and 10.1–4 for the linear adaptive equalizer.

10.3
ADAPTIVE EQUALIZATION OF TRELLIS-CODED SIGNALS

Bandwidth efficient trellis-coded modulation that was described in Section 8.12 is frequently used in digital communications over telephone channels to reduce the required SNR per bit for achieving a specified error rate. Channel distortion of the trellis-coded signal forces us to use adaptive equalization in order to reduce the intersymbol interference. The output of the equalizer is then fed to the Viterbi decoder, which performs soft-decision decoding of the trellis-coded signal.

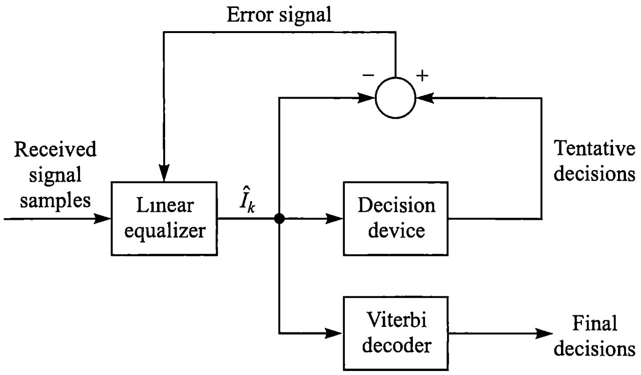

Error signal
−
+
Received signal samples
Linear equalizer
$\hat{I}_k$
Decision device
Tentative decisions
Viterbi decoder
Final decisions

FIGURE 10.3–1
Adjustment of equalizer based on tentative decisions.

The question that arises regarding such a receiver is, how do we adapt the equalizer in a data transmission mode? One possibility is to have the equalizer make its own decisions at its output solely for the purpose of generating an error signal for adjusting its tap coefficients, as shown in the block diagram in Figure 10.3–1. The problem with this approach is that such decisions are generally unreliable, since the pre-decoding coded symbol SNR is relatively low. A high error rate would cause a significant degradation in the operation of the equalizer, which would ultimately affect the reliability of the decisions at the output of the decoder. The more desirable alternative is to use the post-decoding decisions from the Viterbi decoder, which are much more reliable, to continuously adapt the equalizer. This approach is certainly preferable and viable when a linear equalizer is used prior to the Viterbi decoder. The decoding delay inherent in the Viterbi decoder can be overcome by introducing an identical delay in the tap weight adjustment of the equalizer coefficients as shown in Figure 10.3–2. The major price that must be paid for the added delay is that the step-size parameter in the LMS algorithm must be reduced, as described by Long et al. (1987, 1989), in order to achieve stability in the algorithm.

In channels with severe ISI, the linear equalizer is no longer adequate for compensating the channel intersymbol interference. Instead, we would like to use a DFE. But the DFE requires reliable decisions in its feedback filter in order to cancel out the intersymbol interference from previously detected symbols. Tentative decisions prior to decoding would be highly unreliable and, hence, inappropriate. Unfortunately,

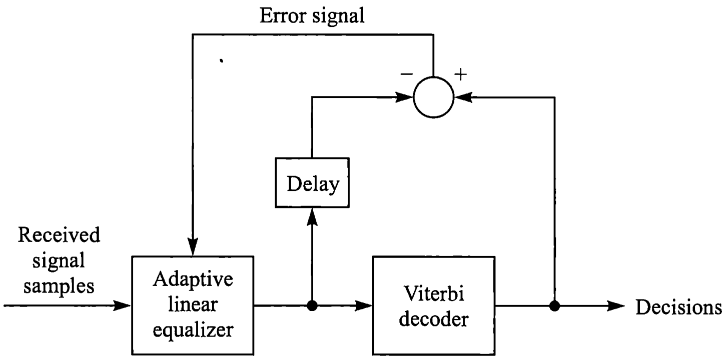

Error signal
−
+
Delay
Received signal samples
Adaptive linear equalizer
Viterbi decoder
Decisions

FIGURE 10.3–2
Adjustment of equalizer based on decisions from the Viterbi decoder.

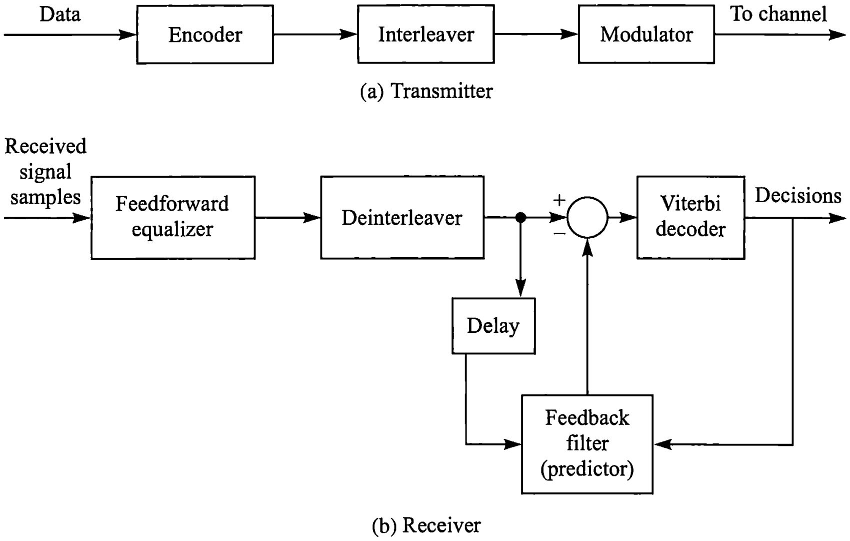


FIGURE 10.3–3
Use of predictive DFE with interleaving and trellis-coded modulation.

the conventional DFE cannot be cascaded with the Viterbi algorithm in which post-decoding decisions from the decoder are fed back to the DFE.

One alternative is to use the predictive DFE described in Section 9.5–3. In order to accommodate for the decoding delay as it affects the linear predictor, we introduce a periodic interleaver/deinterleaver pair that has the same delay as the Viterbi decoder and, thus, makes it possible to generate the appropriate error signal to the predictor as illustrated in the block diagram of Figure 10.3–3. The way in which a predictive DFE can be combined with Viterbi decoding to equalize trellis-coded signals is described and analyzed by Eyuboglu (1988). This same idea has been carried over to the equalization of fading multipath channels by Zhou et al. (1988, 1990), but the structure of the DFE was modified to use recursive least-squares lattice-type filters, which provide faster adaptation to the time variations encountered in the channel.

Another approach that is effective in wireline channels, where the channel impulse response is essentially time invariant, is to place the feedback section of the DFE at the transmitter and, thus, eliminate the tail (postcursors) of the channel response prior to transmission. This is the approach previously described in Section 9.5–4, in which the information sequence is precoded using the Tomlinson–Harashima precoding scheme. Generally, this approach is implemented by sending a channel probe signal to measure the channel frequency or impulse response at the receiver and, thus, to inform the transmitter of the channel response in order to synthesize the precoder. An adaptive, fractionally spaced linear equalizer is implemented at the receiver, which serves as the feedforward filter of the DFE and, thus, compensates for any small time variations in the channel response.

Reduced-state Viterbi detection algorithms From a performance viewpoint, the best method for detecting a TCM signal sequence that is corrupted by ISI is to model the ISI and the trellis code jointly by a single finite state machine and to use the

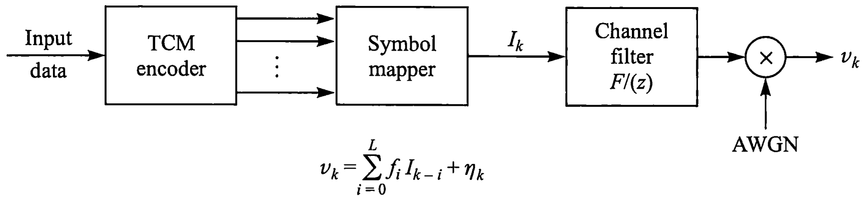


FIGURE 10.3–4
Model of TCM and ISI channel.

Viterbi algorithm on the combined trellis, as described in the papers by Chevillat and Eleftheriou (1988, 1989), Eyuboglu et al. (1988, 1989), and Wesolowski (1987b). By using a whitened matched filter (WMF) as described previously for the receiver front end, the model for the combined trellis encoder and ISI channel filter is illustrated in Figure 10.3–4, where the channel filter $F(z)$ is minimum phase. Thus, a TCM encoder that has S states and employs a signal constellation with 2^{m+1} signal points has a combined TCM/ISI trellis that has $S2^{mL}$ states and 2^m transitions (branches) emerging from each state. The states of the combined finite state machine may be denoted as

$$S_n = (I_{n-L}, I_{n-L+1}, \ldots, I_{n-1}, \theta_n) \tag{10.3–1}$$

where $\{I_n\}$ is the information symbol sequence and where θ_n is the encoder state.

The Viterbi decoder operates on the combined ISI and code trellis in the conventional way, by computing the branch metrics

$$\left| v_k - \sum_{i=0}^{L} f_i I_{k-i} \right|^2 \tag{10.3–2}$$

and incrementing the corresponding path metrics.

Clearly, the complexity of the Viterbi detector becomes prohibitively large when the span L of the ISI is large. In such a case, the decoder complexity can be reduced as described in Section 9.6, by truncating the effective channel memory to L_0 terms. With truncation, the combined TCM/ISI trellis has the $S2^{mL_0}$ states

$$S_n^{L_0} = (I_{n-L_0}, I_{n-L_0+1}, \ldots, I_{n-1}, \theta_n) \tag{10.3–3}$$

where $1 \le L_0 \le L$.

Thus, when $L_0 = 1$, the Viterbi algorithm operates directly on the TCM coded trellis and the L ISI terms are estimated and canceled. By selecting $L_0 > 1$, some ISI terms are kept while $L + 1 - L_0$ terms are canceled. To reduce the performance degradation due to tentative decisions in the Viterbi detector, the ISI cancelation is introduced into the branch metric computations using local feedback, as previously described in Section 9.6. Thus, the branch metrics computed in the Viterbi detector take the form

$$\left| v_k - \sum_{i=0}^{L_0-1} f_i I_{k-i} - \sum_{i=L_0}^{L+1} f_i \tilde{I}_{k-i}\left(S_n^{L_0}\right) \right|^2 \tag{10.3–4}$$

where $\tilde{I}_{k-i}\left(S_n^{L_0}\right)$ denotes the estimated ISI term due to the symbols $\{I_{k-i}, L_0 < i < L\}$ involved in the truncation of the ISI based on local feedback.

In the case of an unknown channel characteristic, both the WMF and the channel estimator of $F(z)$ must be determined adaptively. This may be accomplished by adapting a complex-valued baseband FSE for the WMF and the channel estimator described previously in Section 10.1–7. Thus, a training sequence may be used for initial adjustment and decision-directed estimation may continue following the initial training sequence. The LMS algorithm may be used in both the training and decision-directed modes. Simulation results given by Chevillat and Eleftheriou (1989) demonstrate the superior performance of this adaptive WMF/reduced-state Viterbi detector compared to the combination of a linear equalizer followed by a Viterbi detector.

10.4

RECURSIVE LEAST-SQUARES ALGORITHMS FOR ADAPTIVE EQUALIZATION

The LMS algorithm that we described in Sections 10.1 and 10.2 for adaptively adjusting the tap coefficients of a linear equalizer or a DFE is basically a (stochastic) steepest-descent algorithm in which the true gradient vector is approximated by an estimate obtained directly from the data.

The major advantage of the steepest-descent algorithm lies in its computational simplicity. However, the price paid for the simplicity is slow convergence, especially when the channel characteristics result in an autocorrelation matrix $\boldsymbol{\Gamma}$ whose eigenvalues have a large spread, i.e., $\lambda_{\max}/\lambda_{\min} \gg 1$. Viewed in another way, the gradient algorithm has only a single adjustable parameter for controlling the convergence rate, namely, the parameter Δ. Consequently the slow convergence is due to this fundamental limitation. Two simple methods for increasing the convergence rate to some extent were described in Section 10.1–5.

In order to obtain faster convergence, it is necessary to devise more complex algorithms involving additional parameters. In particular, if the matrix $\boldsymbol{\Gamma}$ is $N \times N$ and has eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_N$, we may use an algorithm that contains N parameters—one for each of the eigenvalues. The optimum selection of these parameters to achieve rapid convergence is a topic of this section.

In deriving faster converging algorithms, we shall adopt a least-squares approach. Thus, we shall deal directly with the received data in minimizing the quadratic performance index, whereas previously we minimized the expected value of the squared error. Put simply, this means that the performance index is expressed in terms of a time average instead of a statistical average.

It is convenient to express the recursive least-squares algorithms in matrix form. Hence, we shall define a number of vectors and matrices that are needed in this development. In so doing, we shall change the notation slightly. Specifically, the estimate of the information symbol at time t, where t is an integer, from a linear equalizer is now expressed as

$$\hat{I}(t) = \sum_{j=-K}^{K} c_j(t-1) v_{t-j}$$

By changing the index j on $c_j(t-1)$ to run from $j = 0$ to $j = N-1$ and simultaneously defining

$$y(t) = v_{t+K}$$

the estimate $\hat{I}(t)$ becomes

$$\begin{aligned} \hat{I}(t) &= \sum_{j=0}^{N-1} c_j(t-1)y(t-j) \\ &= \boldsymbol{C}_N^t(t-1)\boldsymbol{Y}_N(t) \end{aligned} \tag{10.4–1}$$

where $\boldsymbol{C}_N(t-1)$ and $\boldsymbol{Y}_N(t)$ are, respectively, the column vectors of the equalizer coefficients $c_j(t-1)$, $j = 0, 1, \ldots, N-1$, and the input signals $y(t-j)$, $j = 0, 1, 2, \ldots, N-1$.

Similarly, in the decision-feedback equalizer, we have tap coefficients $c_j(t)$, $j = 0, 1, \ldots, N-1$, where the first $K_1 + 1$ are the coefficients of the feedforward filter and the remaining $K_2 = N - K_1 - 1$ are the coefficients of the feedback filter. The data in the estimate $\hat{I}(t)$ is $v_{t+K_1}, \ldots, v_{t+1}, \tilde{I}_{t-1}, \ldots, \tilde{I}_{t-K_2}$, where $\tilde{I}_{t-j}$, $1 \le j \le K_2$, denote the decisions on previously detected symbols. In this development, we neglect the effect of decision errors in the algorithms. Hence, we assume that $\tilde{I}_{t-j} = I_{t-j}$, $1 \le j \le K_2$. For notational convenience, we also define

$$y(t-j) = \begin{cases} v_{t+K_1-j} & (0 \le j \le K_1) \\ I_{t+K_1-j} & (K_1 < j \le N-1) \end{cases} \tag{10.4–2}$$

Thus,

$$\begin{aligned} \boldsymbol{Y}_N(t) &= [y(t) \quad y(t-1) \cdots y(t-N+1)]^t \\ &= [v_{t+K_1} \cdots v_{t+1}\, v_t\, I_{t-1} \cdots I_{t-K_2}]^t \end{aligned} \tag{10.4–3}$$

10.4–1 Recursive Least-Squares (Kalman) Algorithm

The recursive least-squares (RLS) estimation of $\hat{I}(t)$ may be formulated as follows. Suppose we have observed the vectors $\boldsymbol{Y}_N(n)$, $n = 0, 1, \ldots, t$, and we wish to determine the coefficient vector $\boldsymbol{C}_N(t)$ of the equalizer (linear or decision-feedback) that minimizes the time-average weighted squared error

$$\mathcal{E}_N^{LS} = \sum_{n=0}^{t} w^{t-n} |e_N(n, t)|^2 \tag{10.4–4}$$

where the error is defined as

$$e_N(n, t) = I(n) - \boldsymbol{C}_N^t(t)\boldsymbol{Y}_N(n) \tag{10.4–5}$$

and w represents a weighting factor $0 < w < 1$. Thus we introduce exponential weighting into past data, which is appropriate when the channel characteristics are time-variant. Minimization of $\mathcal{E}_N^{LS}$ with respect to the coefficient vector $\boldsymbol{C}_N(t)$ yields the set of linear equations

$$\boldsymbol{R}_N(t)\boldsymbol{C}_N(t) = \boldsymbol{D}_N(t) \tag{10.4–6}$$

where $\boldsymbol{R}_N(t)$ is the signal correlation matrix defined as

$$\boldsymbol{R}_N(t) = \sum_{n=0}^{t} w^{t-n}\boldsymbol{Y}_N^*(n)\boldsymbol{Y}_N^t(n) \tag{10.4–7}$$

and $\boldsymbol{D}_N(t)$ is the cross-correlation vector

$$\boldsymbol{D}_N(t) = \sum_{n=0}^{t} w^{t-n} I(n)\boldsymbol{Y}_N^*(n) \tag{10.4–8}$$

The solution of Equation 10.4–6 is

$$\boldsymbol{C}_N(t) = \boldsymbol{R}_N^{-1}(t)\boldsymbol{D}_N(t) \tag{10.4–9}$$

The matrix $\boldsymbol{R}_N(t)$ is akin to the statistical autocorrelation matrix $\boldsymbol{\Gamma}$, while the vector $\boldsymbol{D}_N(t)$ is akin to the cross-correlation vector $\boldsymbol{\xi}$, defined previously. We emphasize, however, that $\boldsymbol{R}_N(t)$ is not a Toeplitz matrix. We also should mention that, for small values of t, $\boldsymbol{R}_N(t)$ may be ill conditioned; hence, it is customary to initially add the matrix $\delta\boldsymbol{I}_N$ to $\boldsymbol{R}_N(t)$, where δ is a small positive constant and $\boldsymbol{I}_N$ is the identity matrix. With exponential weighting into the past, the effect of adding $\delta\boldsymbol{I}_N$ dissipates with time.

Now suppose we have the solution in Equation 10.4–9 for time $t-1$, i.e., $\boldsymbol{C}_N(t-1)$, and we wish to compute $\boldsymbol{C}_N(t)$. It is inefficient, and, hence, impractical to solve the set of N linear equations for each new signal component that is received. To avoid this, we proceed as follows. First, $\boldsymbol{R}_N(t)$ may be computed recursively as

$$\boldsymbol{R}_N(t) = w\boldsymbol{R}_N(t-1) + \boldsymbol{Y}_N^*(t)\boldsymbol{Y}_N^t(t) \tag{10.4–10}$$

We call Equation 10.4–10 the *time-update equation* for $\boldsymbol{R}_N(t)$.

Since the inverse of $\boldsymbol{R}_N(t)$ is needed in Equation 10.4–9, we use the matrix-inverse identity

$$\boldsymbol{R}_N^{-1}(t) = \frac{1}{w}\left[\boldsymbol{R}_N^{-1}(t-1) - \frac{\boldsymbol{R}_N^{-1}(t-1)\boldsymbol{Y}_N^*(t)\boldsymbol{Y}_N^t(t)\boldsymbol{R}_N^{-1}(t-1)}{w + \boldsymbol{Y}_N^t(t)\boldsymbol{R}_N^{-1}(t-1)\boldsymbol{Y}_N^*(t)}\right] \tag{10.4–11}$$

Thus $\boldsymbol{R}_N^{-1}(t)$ may be computed recursively according to Equation 10.4–11.

For convenience, we define $\boldsymbol{P}_N(t) = \boldsymbol{R}_N^{-1}(t)$. It is also convenient to define an N-dimensional vector, called the *Kalman gain vector*, as

$$\boldsymbol{K}_N(t) = \frac{1}{w + \mu_N(t)}\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t) \tag{10.4–12}$$

where $\mu_N(t)$ is a scalar defined as

$$\mu_N(t) = \boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t) \tag{10.4–13}$$

With these definitions, Equation 10.4–11 becomes

$$\boldsymbol{P}_N(t) = \frac{1}{w}[\boldsymbol{P}_N(t-1) - \boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)] \tag{10.4–14}$$

Suppose we postmultiply both sides of Equation 10.4–14 by $\boldsymbol{Y}_N^*(t)$. Then

$$\begin{aligned}\boldsymbol{P}_N(t)\boldsymbol{Y}_N^*(t) &= \frac{1}{w}[\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t) - \boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t)]\\ &= \frac{1}{w}\{[w+\mu_N(t)]\boldsymbol{K}_N(t) - \boldsymbol{K}_N(t)\mu_N(t)\}\\ &= \boldsymbol{K}_N(t)\end{aligned} \tag{10.4–15}$$

Therefore, the Kalman gain vector may also be defined as $\boldsymbol{P}_N(t)\boldsymbol{Y}_N(t)$.

Now we use the matrix inversion identity to derive an equation for obtaining $\boldsymbol{C}_N(t)$ from $\boldsymbol{C}_N(t-1)$. Since

$$\boldsymbol{C}_N(t) = \boldsymbol{P}_N(t)\boldsymbol{D}_N(t)$$

and

$$\boldsymbol{D}_N(t) = w\boldsymbol{D}_N(t-1) + I(t)\boldsymbol{Y}_N^*(t) \tag{10.4–16}$$

we have

$$\begin{aligned}\boldsymbol{C}_N(t) &= \frac{1}{w}[\boldsymbol{P}_N(t-1) - \boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)][w\boldsymbol{D}_N(t-1) + I(t)\boldsymbol{Y}_N^*(t)]\\ &= \boldsymbol{P}_N(t-1)\boldsymbol{D}_N(t-1) + \frac{1}{w}I(t)\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t)\\ &\quad - \boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)\boldsymbol{D}_N(t-1)\\ &\quad - \frac{1}{w}I(t)\boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t)\\ &= \boldsymbol{C}_N(t-1) + \boldsymbol{K}_N(t)[I(t) - \boldsymbol{Y}_N^t(t)\boldsymbol{C}_N(t-1)]\end{aligned} \tag{10.4–17}$$

Note that $\boldsymbol{Y}_N^t(t)\boldsymbol{C}_N(t-1)$ is the output of the equalizer at time t, i.e.,

$$\hat{I}(t) = \boldsymbol{Y}_N^t(t)\boldsymbol{C}_N(t-1) \tag{10.4–18}$$

and

$$e_N(t, t-1) = I(t) - \hat{I}(t) \equiv e_N(t) \tag{10.4–19}$$

is the error between the desired symbol and the estimate. Hence, $\boldsymbol{C}_N(t)$ is updated recursively according to the relation

$$\boldsymbol{C}_N(t) = \boldsymbol{C}_N(t-1) + \boldsymbol{K}_N(t)e_N(t) \tag{10.4–20}$$

The residual MSE resulting from this optimization is

$$\mathcal{E}_{N\min}^{LS} = \sum_{m=0}^{t} w^{t-n}|I(n)|^2 - \boldsymbol{C}_N^t(t)\boldsymbol{D}_N^*(t) \tag{10.4–21}$$

To summarize, suppose we have $\boldsymbol{C}_N(t-1)$ and $\boldsymbol{P}_N(t-1)$. When a new signal component is received, we have $\boldsymbol{Y}_N(t)$. Then the recursive computation for the time update of $\boldsymbol{C}_N(t)$ and $\boldsymbol{P}_N(t)$ proceeds as follows:

- Compute output:

$$\hat{I}(t) = \boldsymbol{Y}_N^t(t)\boldsymbol{C}_N(t-1)$$

- Compute error:

$$e_N(t) = I(t) - \hat{I}(t)$$

- Compute Kalman gain vector:

$$\boldsymbol{K}_N(t) = \frac{\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^t(t)}{w + \boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)\boldsymbol{Y}_N^*(t)}$$

- Update inverse of the correlation matrix:

$$\boldsymbol{P}_N(t) = \frac{1}{w}[\boldsymbol{P}_N(t-1) - \boldsymbol{K}_N(t)\boldsymbol{Y}_N^t(t)\boldsymbol{P}_N(t-1)]$$

- Update coefficients:

$$\begin{aligned}\boldsymbol{C}_N(t) &= \boldsymbol{C}_N(t-1) + \boldsymbol{K}_N(t)e_N(t)\\ &= \boldsymbol{C}_N(t-1) + \boldsymbol{P}_N(t)\boldsymbol{Y}_N^*(t)e_N(t)\end{aligned} \tag{10.4–22}$$

The algorithm described by Equation 10.4–22 is called the *RLS direct form* or *Kalman algorithm.* It is appropriate when the equalizer has a transversal (direct-form) structure.

Note that the equalizer coefficients change with time by an amount equal to the error $e_N(t)$ multipled by the Kalman gain vector $\boldsymbol{K}_N(t)$. Since $\boldsymbol{K}_N(t)$ is N-dimensional, each tap coefficient in effect is controlled by one of the elements of $\boldsymbol{K}_N(t)$. Consequently rapid convergence is obtained. In contrast, the steepest-descent algorithm, expressed in our present notation, is

$$\boldsymbol{C}_N(t) = \boldsymbol{C}_N(t-1) + \Delta\boldsymbol{Y}_N^*(t)e_N(t) \tag{10.4–23}$$

and the only variable parameter is the step size Δ.

Figure 10.4–1 illustrates the initial convergence rate of these two algorithms for a channel with fixed parameters $f_0 = 0.26$, $f_1 = 0.93$, $f_2 = 0.26$, and a linear equalizer with 11 taps. The eigenvalue ratio for this channel is $\lambda_{\max}/\lambda_{\min} = 11$. All the equalizer coefficients were initialized to zero. The steepest-descent algorithm was implemented with $\Delta = 0.020$. The superiority of the Kalman algorithm is clearly evident. This is especially important in a time-variant channel. For example, the time variations in the characteristics of an (ionospheric) high-frequency (HF) radio channel are too rapid to be equalized by the gradient algorithm, but the Kalman algorithm adapts sufficiently rapidly to track such variations.

In spite of its superior convergence performance, the Kalman algorithm described above has two disadvantages. One is its complexity. The second is its sensitivity to

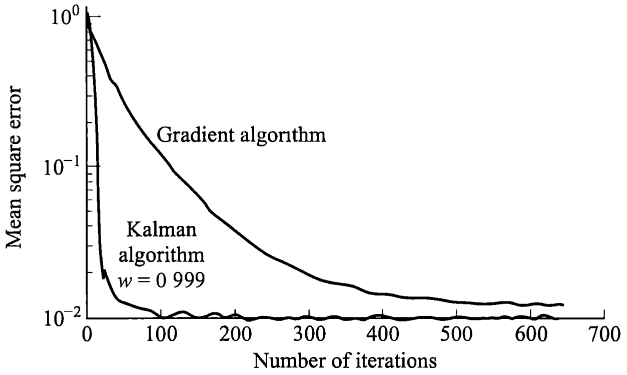


FIGURE 10.4–1
Comparison of convergence rate for the Kalman and gradient algorithms.

roundoff noise that accumulates due to the recursive computations. The latter may cause instabilities in the algorithm.

The number of computations or operations (multiplications, divisions, and subtractions) in computing the variables in Equation 10.4–22 is proportional to N^2. Most of these operations are involved in the updating of $\boldsymbol{P}_N(t)$. This part of the computation is also susceptible to roundoff noise. To remedy that problem, algorithms have been developed that avoid the computation of $\boldsymbol{P}_N(t)$ according to Equation 10.4–14. The basis of these algorithms lies in the decomposition of $\boldsymbol{P}_N(t)$ in the form

$$\boldsymbol{P}_N(t) = \boldsymbol{S}_N(t)\boldsymbol{\Lambda}_N(t)\boldsymbol{S}'_N(t) \tag{10.4–24}$$

where $\boldsymbol{S}_N(t)$ is a lower-triangular matrix whose diagonal elements are unity, and $\boldsymbol{\Lambda}_N(t)$ is a diagonal matrix. Such a decomposition is called a *square-root factorization* (see Bierman, 1977). This factorization is described in Appendix D. In a square-root algorithm, $\boldsymbol{P}_N(t)$ is not updated as in Equation 10.4–14 nor is it computed. Instead, the time updating is performed on $\boldsymbol{S}_N(t)$ and $\boldsymbol{\Lambda}_N(t)$.

Square-root algorithms are frequently used in control systems applications in which Kalman filtering is involved. In digital communications, the square-root Kalman algorithm has been implemented in a decision-feedback-equalized PSK modem designed to transmit at high speed over high-frequency radio channels with a nominal 3-kHz bandwidth. This algorithm is described in the paper by Hsu (1982). It has a computational complexity of $1.5N^2 + 6.5N$ (complex-valued multiplications and divisions per output symbol). It is also numerically stable and exhibits good numerical properties. For a detailed discussion of square-root algorithms in sequential estimation, the reader is referred to the book by Bierman (1977).

It is also possible to derive RLS algorithms with computational complexities that grow linearly with the number N of equalizer coefficients. Such algorithms are generally called *fast RLS algorithms* and have been described in the papers by Carayannis et al. (1983), Cioffi and Kailath (1984), and Slock and Kailath (1991).

Another class of recursive least squares algorithms for adaptive equalization are based on the lattice equalizer structure. Below, we derive the lattice filter structure from the transversal filter structure and, thus, demonstrate the equivalence of the two structures.

10.4–2 Linear Prediction and the Lattice Filter

In this section we develop the connection between a linear FIR filter and a lattice filter. This connection is most easily established by considering the problem of linear prediction of a signal sequence.

The linear prediction problem may be stated as follows: given a set of data $y(t-1), y(t-2), \ldots, y(t-p)$, predict the value of the next data point $y(t)$. The predictor of order p is

$$\hat{y}(t) = \sum_{k=1}^{p} a_{pk} y(t-k) \tag{10.4–25}$$

Minimization of the MSE, defined as

$$\begin{aligned} \mathcal{E}_p &= E[y(t) - \hat{y}(t)]^2 \\ &= E\left[y(t) - \sum_{k=1}^{p} a_{pk} y(t-k) \right]^2 \end{aligned} \tag{10.4–26}$$

with respect to the predictor coefficients $\{a_{pk}\}$ yields the set of linear equations

$$\sum_{k=1}^{p} a_{pk} R(k-l) = R(l), \qquad l = 1, 2, \ldots, p \tag{10.4–27}$$

where

$$R(l) = E[y(t)y(t+l)]$$

These are called the *normal equations* or the *Yule–Walker equations.*

The matrix $\boldsymbol{R}$ with elements $\boldsymbol{R}(k-l)$ is a Toeplitz matrix, and, hence, the Levinson–Durbin algorithm provides an efficient means for solving the linear equations recursively, starting with a first-order predictor and proceeding recursively to the solution of the coefficients for the predictor of order p. The recursive relations for the Levinson–Durbin algorithm are (see Levinson (1947) and Durbin (1959))

$$\begin{aligned} a_{11} &= \frac{R(1)}{R(0)}, \qquad \mathcal{E}_0 = R(0) \\ a_{mm} &= \frac{\phi(m) - \boldsymbol{A}_m^t \boldsymbol{R}_{m-1}^r}{\mathcal{E}_{m-1}} \\ a_{mk} &= a_{m-1k} - a_{mm} a_{m-1m-k} \\ \mathcal{E}_m &= \mathcal{E}_{m-1}\left(1 - a_{mm}^2\right) \end{aligned} \tag{10.4–28}$$

for $m = 1, 2, \ldots, p$, where the vectors $\boldsymbol{A}_{m-1}$ and $\boldsymbol{R}_{m-1}^r$ are defined as

$$\begin{aligned} \boldsymbol{A}_{m-1} &= [a_{m-11} \quad a_{m-12} \cdots a_{m-1m-1}]^t \\ \boldsymbol{R}_{m-1}^r &= [R(m-1) \quad R(m-2) \cdots R(1)]^t \end{aligned}$$

The linear prediction filter of order m may be realized as a transversal (FIR) filter with transfer function

$$A_m(z) = 1 - \sum_{k=1}^{m} a_m z^{-k} \tag{10.4–29}$$

Its input is the data $\{y(t)\}$ and its output is the error $e(t) = y(t) - \hat{y}(t)$. The prediction filter can also be realized in the form of a lattice, as we now demonstrate.

Our starting point is the use of the Levinson–Durbin algorithm for the predictor coefficients a_{mk} in Equation 10.4–29. This substitution yields

$$\begin{aligned} A_m(z) &= 1 - \sum_{k=1}^{m-1} (a_{m-1k} - a_{mm} a_{m-1m-k}) z^{-k} - a_{mm} z^{-m} \\ &= 1 - \sum_{k=1}^{m-1} a_{m-1k} z^{-k} - a_{mm} z^{-m} \left(1 - \sum_{k=1}^{m-1} a_{m-1k} z^{k} \right) \\ &= A_{m-1}(z) - a_{mm} z^{-m} A_{m-1}(z^{-1}) \end{aligned} \tag{10.4–30}$$

Thus we have the transfer function of the mth-order predictor in terms of the transfer function of the $(m-1)$th-order predictor.

Now suppose we define a filter with transfer function $G_m(z)$ as

$$G_m(z) = z^{-m} A_m(z^{-1}) \tag{10.4–31}$$

Then Equation 10.4–30 may be expressed as

$$A_m(z) = A_{m-1}(z) - a_{mm} z^{-1} G_{m-1}(z) \tag{10.4–32}$$

Note that $G_{m-1}(z)$ represents a transversal filter with tap coefficients $(-a_{m-1m-1}, -a_{m-1m-2}, \ldots, -a_{m-11}, 1)$, while the coefficients of $A_{m-1}(z)$ are exactly the same except that they are given in reverse order.

More insight into the relationship between $A_m(z)$ and $G_m(z)$ can be obtained by computing the output of these two filters to an input sequence $y(t)$. Using z-transform relations, we have

$$A_m(z)Y(z) = A_{m-1}(z)Y(z) - a_{mm} z^{-1} G_{m-1}(z)Y(z) \tag{10.4–33}$$

We define the outputs of the filters as

$$\begin{aligned} F_m(z) &= A_m(z)Y(z) \\ B_m(z) &= G_m(z)Y(z) \end{aligned} \tag{10.4–34}$$

Then Equation 10.4–33 becomes

$$F_m(z) = F_{m-1}(z) - a_{mm} z^{-1} B_{m-1}(z) \tag{10.4–35}$$

In the time domain, the relation in Equation 10.4–35 becomes

$$f_m(t) = f_{m-1}(t) - a_{mm} b_{m-1}(t-1), \qquad m \geq 1 \tag{10.4–36}$$

where

$$f_m(t) = y(t) - \sum_{k=1}^{m-1} a_{mk} y(t-k) \tag{10.4–37}$$

$$b_m(t) = y(t-m) - \sum_{k=1}^{m-1} a_{mk} y(t-m+k) \tag{10.4–38}$$

To elaborate, $f_m(t)$ in Equation 10.4–37 represents the error of an mth-order forward predictor, while $b_m(t)$ represents the error of an mth-order backward predictor.

The relation in Equation 10.4–36 is one of two that specifies a lattice filter. The second relation is obtained from $G_m(z)$ as follows:

$$\begin{aligned} G_m(z) &= z^{-m} A_m(z^{-1}) \\ &= z^{-m}[A_{m-1}(z^{-1}) - a_{mm} z^m A_{m-1}(z)] \\ &= z^{-1} G_{m-1}(z) - a_{mm} A_{m-1}(z) \end{aligned} \tag{10.4–39}$$

Now, if we multiply both sides of Equation 10.4–39 by $Y(z)$ and express the result in terms of $F_m(z)$ and $B_m(z)$ using the definitions in Equation 10.4–34, we obtain

$$B_m(z) = z^{-1} B_{m-1}(z) - a_{mm} F_{m-1}(z) \tag{10.4–40}$$

By transforming Equation 10.4–40 into the time domain, we obtain the second relation that corresponds to the lattice filter, namely,

$$b_m(t) = b_{m-1}(t-1) - a_{mm} f_{m-1}(t), \qquad m \geq 1 \tag{10.4–41}$$

The initial condition is

$$f_0(t) = b_0(t) = y(t) \tag{10.4–42}$$

The lattice filter described by the recursive relations in Equations 10.4–36 and 10.4–41 is illustrated in Figure 10.4–2. Each stage is characterized by its own multiplication factor $\{a_{ii}\}$, $i = 1, 2, \ldots, m$, which is defined in the Levinson–Durbin algorithm. The forward and backward errors $f_m(t)$ and $b_m(t)$ are usually called the *residuals*. The mean square value of these residuals is

$$\mathcal{E}_m = E\left[f_m^2(t)\right] = E\left[b_m^2(t)\right] \tag{10.4–43}$$

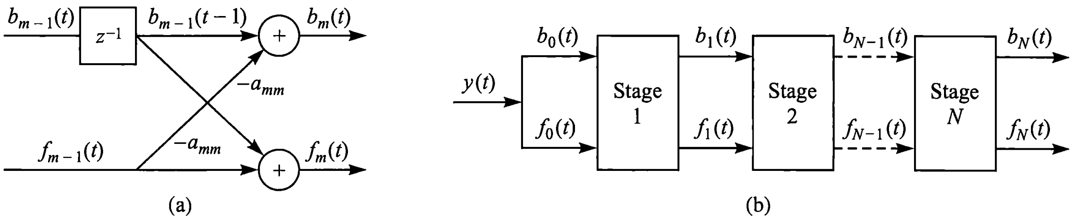


FIGURE 10.4–2
A lattice filter.

$\mathcal{E}_m$ is given recursively, as indicated in the Levinson–Durbin algorithm, by

$$\begin{aligned}\mathcal{E}_m &= \mathcal{E}_{m-1}\left(1 - a_{mm}^2\right) \\ &= \mathcal{E}_0 \prod_{i=1}^{m} \left(1 - a_{ii}^2\right)\end{aligned} \tag{10.4–44}$$

where $\mathcal{E}_0 = R(0)$.

The residuals $\{f_m(t)\}$ and $\{b_m(t)\}$ satisfy a number of interesting properties, as described by Makhoul (1978). Most important of these are the orthogonality properties

$$\begin{aligned}E[b_m(t)b_n(t)] &= \mathcal{E}_m \delta_{mn} \\ E[f_m(t+m)f_n(t+n)] &= \mathcal{E}_m \delta_{mn}\end{aligned} \tag{10.4–45}$$

Furthermore, the cross correlation between $f_m(t)$ and $b_n(t)$ is

$$E[f_m(t)b_n(t)] = \begin{cases} a_{nn}\mathcal{E}_m & m \geq n \\ 0 & m < n \end{cases} \qquad m, n \geq 0 \tag{10.4–46}$$

As a consequence of the orthogonality properties of the residuals, the different sections of the lattice exhibit a form of independence that allows us to add or delete one or more of the last stages without affecting the parameters of the remaining stages. Since the residual mean square error $\mathcal{E}_m$ decreases monotonically with the number of sections, $\mathcal{E}_m$ can be used as a performance index in determining where the lattice should be terminated.

From the above discussion, we observe that a linear prediction filter can be implemented either as a linear transversal filter or as a lattice filter. The lattice filter is order-recursive, and, as a consequence, the number of sections it contains can be easily increased or decreased without affecting the parameters of the remaining sections. In contrast, the coefficients of a transversal filter obtained on the basis of the RLS criterion are interdependent. This means that an increase or a decrease in the size of the filter results in a change in all coefficients. Consequently, the Kalman algorithm described in Section 10.4–1 is recursive in time but not in order.

Based on least-squares optimization, RLS lattice equalization algorithms have been developed whose computational complexity grows linearly with the number N of filter coefficients (lattice stages). Hence, the lattice equalizer structure is computationally competitive with the direct-form fast RLS equalizer algorithms. For example, Figure 10.4–3 illustrates the computational complexity (number of multiplications and divisions per output symbol) of transversal and lattice, symbol-spaced DFE filter structures. Observe that for equalizer lengths of fewer than 10 taps, the difference in computational complexity among the different structures and algorithms is relatively small. However, as the number of taps increases, the lattice RLS algorithm and the fast (transversal) RLS algorithm are significantly less complex than the conventional and square-root RLS algorithms. Of course, all the RLS algorithms are computationally more complex than the LMS algorithm. RLS lattice algorithms are described in the papers by Morf (1977), Morf and Lee (1978), and Morf et al. (1977a,b,c), Satorius and Alexander (1979), Satorius and Pack (1981), Ling and Proakis (1982, 1984c, 1985, 1986) and in the books by Proakis et al. (2002) and Haykin (2002).

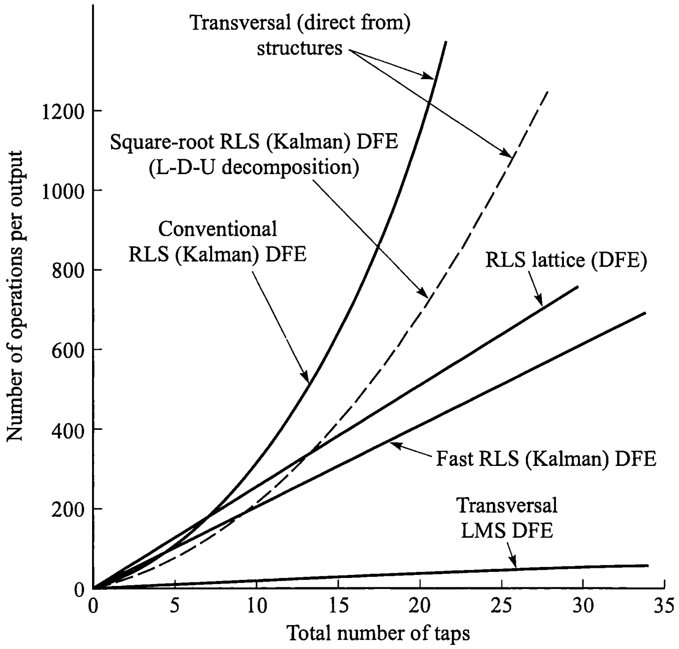


FIGURE 10.4–3
Computational complexity of DFE algorithms.

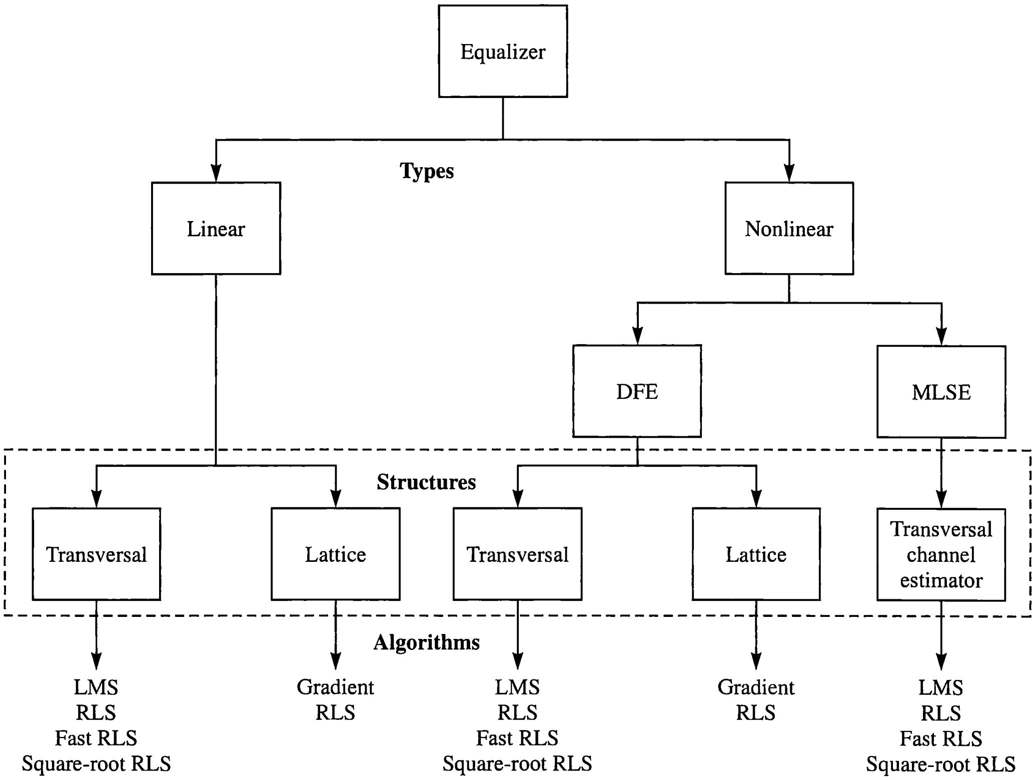

Equalizer
Types
Linear
Nonlinear
DFE
MLSE
Structures
Transversal
Lattice
Transversal
Lattice
Transversal
channel
estimator
Algorithms
LMS
RLS
Fast RLS
Square-root RLS
Gradient
RLS
LMS
RLS
Fast RLS
Square-root RLS
Gradient
RLS
LMS
RLS
Fast RLS
Square-root RLS

FIGURE 10.4–4
Equalizer types, structures, and algorithms.

RLS lattice algorithms have the distinct feature of being numerically robust to round-off error inherent in digital implementations of the algorithms. A treatment of their numerical properties may be found in the papers by Ling and Proakis (1984a) and Ling et al. (1986a,b).

Figure 10.4–4 illustrater the different types of linear and nonlinear equalizers the corresponding structures for their implementation, and the adaptive algorithms that may be used to adjust the equalizer coefficients.

10.5
SELF-RECOVERING (BLIND) EQUALIZATION

In the conventional zero-forcing or minimum MSE equalizers, we assumed that a known training sequence is transmitted to the receiver for the purpose of initially adjusting the equalizer coefficients. However, there are some applications, such as multipoint communication networks, where it is desirable for the receiver to synchronize to the received signal and to adjust the equalizer without having a known training sequence available. Equalization techniques based on initial adjustment of the coefficients without the benefit of a training sequence are said to be *self-recovering* or *blind*.

Beginning with the paper by Sato (1975), three different classes of adaptive blind equalization algorithms have been developed over the past three decades. One class of algorithms is based on steepest descent for adaptation of the equalizer. A second class of algorithms is based on the use of second- and higher-order (generally, fourth-order) statistics of the received signal to estimate the channel characteristics and to design the equalizer. More recently, a third class of blind equalization algorithms based on the maximum-likelihood criterion have been investigated. In this section, we briefly describe these approaches and give several relevant references to the literature.

10.5–1 Blind Equalization Based on the Maximum-Likelihood Criterion

It is convenient to use the equivalent, discrete-time channel model described in Section 9.3–2. Recall that the output of this channel model with ISI is

$$.v_n = \sum_{k=0}^{L} f_k I_{n-k} + \eta_n \tag{10.5–1}$$

where $\{f_k\}$ are the equivalent discrete-time channel coefficients, $\{I_n\}$ represents the information sequence, and $\{\eta_n\}$ is a white Gaussian noise sequence.

For a block of N received data points, the (joint) probability density function of the received data vector $\boldsymbol{v} = [v_1 \; v_2 \; \cdots \; v_N]^t$ conditioned on knowing the impulse response vector $\boldsymbol{f} = [f_0 \; f_1 \; \cdots \; f_L]^t$ and the data vector $\boldsymbol{I} = [I_1 \; I_2 \; \cdots I_N]^t$ is

$$p(\boldsymbol{v}|\boldsymbol{f}, \boldsymbol{I}) = \frac{1}{(2\pi\sigma^2)^N} \exp\left(-\frac{1}{2\sigma^2}\sum_{n=1}^{N}\left|v_n - \sum_{k=0}^{L} f_k I_{n-k}\right|^2\right) \tag{10.5–2}$$

The joint maximum-likelihood estimates of $\boldsymbol{f}$ and $\boldsymbol{I}$ are the values of these vectors that maximize the joint probability density function $p(\boldsymbol{v}|\boldsymbol{f}, \boldsymbol{I})$ or, equivalently, the values of $\boldsymbol{f}$ and $\boldsymbol{I}$ that minimize the term in the exponent. Hence, the ML solution is simply the minimum over $\boldsymbol{f}$ and $\boldsymbol{I}$ of the metric

$$\begin{aligned} DM(\boldsymbol{I}, \boldsymbol{f}) &= \sum_{n=1}^{N} \left| v_n - \sum_{k=0}^{L} f_k I_{n-k} \right|^2 \\ &= \|\boldsymbol{v} - \boldsymbol{A}\boldsymbol{f}\|^2 \end{aligned} \tag{10.5–3}$$

where the matrix $\boldsymbol{A}$ is called the *data matrix* and is defined as

$$\boldsymbol{A} = \begin{bmatrix} I_1 & 0 & 0 & \dots & 0 \\ I_2 & I_1 & 0 & \dots & 0 \\ I_3 & I_2 & I_1 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ I_N & I_{N-1} & I_{N-2} & \dots & I_{N-L} \end{bmatrix} \tag{10.5–4}$$

We make several observations. First of all, we note that when the data vector $\boldsymbol{I}$ (or the data matrix $\boldsymbol{A}$) is known, as is the case when a training sequence is available at the receiver, the ML channel impulse response estimate obtained by minimizing Equation 10.5–3 over $\boldsymbol{f}$ is

$$\boldsymbol{f}_{ML}(\boldsymbol{I}) = (\boldsymbol{A}^H \boldsymbol{A})^{-1} \boldsymbol{A}^H \boldsymbol{v} \tag{10.5–5}$$

On the other hand, when the channel impulse response $\boldsymbol{f}$ is known, the optimum ML detector for the data sequence $\boldsymbol{I}$ performs a trellis search (or tree search) by utilizing the Viterbi algorithm for the ISI channel.

When neither $\boldsymbol{I}$ nor $\boldsymbol{f}$ are known, the minimization of the performance index $DM(\boldsymbol{I}, \boldsymbol{f})$ may be performed jointly over $\boldsymbol{I}$ and $\boldsymbol{f}$. Alternatively, $\boldsymbol{f}$ may be estimated from the probability density function $p(\boldsymbol{v}|\boldsymbol{f})$, which may be obtained by averaging $p(\boldsymbol{v}, \boldsymbol{f}|\boldsymbol{I})$ over all possible data sequences. That is,

$$\begin{aligned} p(\boldsymbol{v}|\boldsymbol{f}) &= \sum_m p(\boldsymbol{v}, \boldsymbol{I}^{(m)}|\boldsymbol{f}) \\ &= \sum_m p(\boldsymbol{v}|\boldsymbol{I}^{(m)}, \boldsymbol{f}) P(\boldsymbol{I}^{(m)}) \end{aligned} \tag{10.5–6}$$

where $P(\boldsymbol{I}^{(m)})$ is the probability of the sequence $\boldsymbol{I} = \boldsymbol{I}^{(m)}$, for $m = 1, 2, \dots, M^N$, and M is the size of the signal constellation.

Channel estimation based on average over data sequences As indicated in the above discussion, when both $\boldsymbol{I}$ and $\boldsymbol{f}$ are unknown, one approach is to estimate the impulse response $\boldsymbol{f}$ after averaging the probability density $p(\boldsymbol{v}, \boldsymbol{I}|\boldsymbol{f})$ over all possible data sequences. Thus, we have

$$\begin{aligned} p(\boldsymbol{v}|\boldsymbol{f}) &= \sum_m p(\boldsymbol{v}|\boldsymbol{I}^{(m)}, \boldsymbol{f})P(\boldsymbol{I}^{(m)}) \\ &= \sum_m \left[\frac{1}{(2\pi\sigma^2)^N}\exp\left(-\frac{\|\boldsymbol{v}-\boldsymbol{A}^{(m)}\boldsymbol{f}\|^2}{2\sigma^2}\right)\right] P(\boldsymbol{I}^{(m)}) \end{aligned} \tag{10.5–7}$$

Then, the estimate of $\boldsymbol{f}$ that maximizes $p(\boldsymbol{v}|\boldsymbol{f})$ is the solution of the equation

$$\begin{aligned} \frac{\partial p(\boldsymbol{v}|\boldsymbol{f})}{\partial \boldsymbol{f}} &= \sum_m P(\boldsymbol{I}^{(m)}) \cdot \\ (\boldsymbol{A}^{(m)H}\boldsymbol{A}^{(m)}\boldsymbol{f} - \boldsymbol{A}^{(m)H}\boldsymbol{v})&\exp\left(-\frac{\|\boldsymbol{v}-\boldsymbol{A}^{(m)}\boldsymbol{f}\|^2}{2\sigma^2}\right) = 0 \end{aligned} \tag{10.5–8}$$

Hence, the estimate of $\boldsymbol{f}$ may be expressed as

$$\begin{aligned} \boldsymbol{f} = &\left[\sum_m P(\boldsymbol{I}^{(m)})\boldsymbol{A}^{(m)H}\boldsymbol{A}^{(m)} g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f})\right]^{-1} \\ &\times \sum_m P(\boldsymbol{I}^{(m)}) g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f})\boldsymbol{A}^{(m)H}\boldsymbol{v} \end{aligned} \tag{10.5–9}$$

where the function $g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f})$ is defined as

$$g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f}) = \exp\left(-\frac{\|\boldsymbol{v}-\boldsymbol{A}^{(m)}\boldsymbol{f}\|^2}{2\sigma^2}\right) \tag{10.5–10}$$

The resulting solution for the optimum $\boldsymbol{f}$ is denoted by $\boldsymbol{f}_{ML}$.

Equation 10.5–9 is a non-linear equation for the estimate of the channel impulse response, given the received signal vector $\boldsymbol{v}$. It is generally difficult to obtain the optimum solution by solving Equation 10.5–9 directly. On the other hand, it is relatively simple to devise a numerical method that solves for $\boldsymbol{f}_{ML}$ recursively. Specifically, we may write

$$\begin{aligned} \boldsymbol{f}^{(k+1)} = &\left[\sum_m P(\boldsymbol{I}^{(m)})\boldsymbol{A}^{(m)H}\boldsymbol{A}^{(m)} g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f}^{(k)})\right]^{-1} \\ &\times \sum_m P(\boldsymbol{I}^{(m)}) g(\boldsymbol{v}, \boldsymbol{A}^{(m)}, \boldsymbol{f}^{(k)})\boldsymbol{A}^{(m)H}\boldsymbol{v} \end{aligned} \tag{10.5–11}$$

Once $\boldsymbol{f}_{ML}$ is obtained from the solution of Equation 10.5–9 or 10.5–11, we may simply use the estimate in the minimization of the metric $DM(\boldsymbol{I}, \boldsymbol{f}_{ML})$, given by Equation 10.5–3, over all the possible data sequences. Thus, $\boldsymbol{I}_{ML}$ is the sequence $\boldsymbol{I}$ that minimizes $DM(\boldsymbol{I}, \boldsymbol{f}_{ML})$, i.e.,

$$\min_{\boldsymbol{I}} DM(\boldsymbol{I}, \boldsymbol{f}_{ML}) = \min_{\boldsymbol{I}} \|\boldsymbol{v} - \boldsymbol{A}\boldsymbol{f}_{ML}\|^2 \tag{10.5–12}$$

We know that the Viterbi algorithm is the computationally efficient algorithm for performing the minimization of $DM(\boldsymbol{I}, \boldsymbol{f}_{ML})$ over $\boldsymbol{I}$.

This algorithm has two major drawbacks. First, the recursion for f_{LM} given by Equation 10.5–11 is computationally intensive. Second, and, perhaps, more importantly, the estimate f_{ML} is not as good as the maximum-likelihood estimate $f_{ML}(I)$ that is obtained when the sequence I is known. Consequently, the error rate performance of the blind equalizer (the Viterbi algorithm) based on the estimate f_{ML} is poorer than that based on $f_{ML}(I)$. Next, we consider joint channel and data estimation.

Joint channel and data estimation Here, we consider the joint optimization of the performance index $DM(I, f)$ given by Equation 10.5–3. Since the elements of the impulse response vector f are continuous and the elements of the data vector I are discrete, one approach is to determine the maximum-likelihood estimate of f for each possible data sequence and, then, to select the data sequence that minimizes $DM(I, f)$ for each corresponding channel estimate. Thus, the channel estimate corresponding to the mth data sequence $I^{(m)}$ is

$$f_{ML}(I^{(m)}) = (A^{(m)t}A^{(m)})^{-1}A^{(m)t}v \tag{10.5–13}$$

For the mth data sequence, the metric $DM(I, f)$ becomes

$$DM\left[I^{(m)}, f_{ML}(I^{(m)})\right] = \|v - A^{(m)}f_{ML}(I^{(m)})\|^2 \tag{10.5–14}$$

Then, from the set of M^N possible sequences, we select the data sequence that minimizes the cost function in Equation 10.5–14, i.e., we determine

$$\min_{I^{(m)}} DM\left[I^{(m)}, f_{ML}(I^{(m)})\right] \tag{10.5–15}$$

The approach described above is an exhaustive computational search method with a computational complexity that grows exponentially with the length of the data block. We may select $N = L + 1$, and, thus, we shall have one channel estimate for each of the M^L surviving sequences. Thereafter, we may continue to maintain a separate channel estimate for each surviving path of the Viterbi algorithm search through the trellis. This approach to joint channel and data estimation has been called *per-survivor processing* by Raheli et al. (1995).

A similar approach has been proposed by Seshadri (1994). In essence, Seshadri's algorithm is a type of generalized Viterbi algorithm (GVA) that retains $K \geq 1$ best estimates of the transmitted data sequence into each state of the trellis and the corresponding channel estimates. In Seshadri's GVA, the search is identical to the conventional Viterbi algorithm (VA) from the beginning up to the Lth stage of the trellis, i.e., up to the point where the received sequence $(v_1, v_2, \ldots, v_L)$ has been processed. Hence, up to the Lth stage, an exhaustive search is performed. Associated with each data sequence $I^{(m)}$, there is a corresponding channel estimate $f_{ML}(I^{(m)})$. From this stage on, the search is modified, to retain $K \geq 1$ surviving sequences and associated channel estimates per state instead of only one sequence per state. Thus, the GVA is used for processing the received signal sequence $\{v_n, n \geq L + 1\}$. The channel estimate is updated recursively at each stage using the LMS algorithm to further reduce the computational complexity. Simulation results given in the paper by Seshadri (1994) indicate that this GVA blind equalization algorithm performs rather well at moderate signal-to-noise ratios with $K = 4$. Hence, there is a modest increase in the computational complexity of the

GVA compared with that for the conventional VA. However, there are additional computations involved with the estimation and updating of the channel estimates $\boldsymbol{f}(\boldsymbol{I}^{(m)})$ associated with each of the surviving data estimates.

An alternative joint estimation algorithm that avoids the least-squares computation for channel estimation has been devised by Zervas et al. (1991). In this algorithm, the order for performing the joint minimization of the performance index $DM(\boldsymbol{I}, \boldsymbol{f})$ is reversed. That is, a channel impulse response, say $\boldsymbol{f} = \boldsymbol{f}^{(1)}$, is selected and then the conventional VA is used to find the optimum sequence for this channel impulse response. Then, we may modify $\boldsymbol{f}^{(1)}$ in some manner to $\boldsymbol{f}^{(2)} = \boldsymbol{f}^{(1)} + \Delta \boldsymbol{f}^{(1)}$ and repeat the optimization over the data sequences $\{\boldsymbol{I}^{(m)}\}$.

Based on this general approach, Zervas et al. developed a new ML blind equalization algorithm, which is called a *quantized-channel algorithm*. The algorithm operates over a grid in the channel space, which becomes finer and finer by using the ML criterion to confine the estimated channel in the neighborhood of the original unknown channel. This algorithm leads to an efficient parallel implementation, and its storage requirements are only those of the VA.

10.5–2 Stochastic Gradient Algorithms

Another class of blind equalization algorithms are stochastic-gradient iterative equalization schemes that apply a memoryless non-linearity in the output of a linear FIR equalization filter in order to generate the "desired response" in each iteration.

Let us begin with an initial guess of the coefficients of the optimum equalizer, which we denote by $\{c_n\}$. Then, the convolution of the channel response with the equalizer response may be expressed as

$$\{c_n\} \star \{f_n\} = \{\delta_n\} + \{e_n\} \tag{10.5–16}$$

where $\{\delta_n\}$ is the unit sample sequence and $\{e_n\}$ denotes the error sequence that results from our initial guess of the equalizer coefficients. If we convolve the equalizer impulse response with the received sequence $\{v_n\}$, we obtain

$$\begin{aligned} \{\hat{I}_n\} &= \{v_n\} \star \{c_n\} \\ &= \{I_n\} \star \{f_n\} \star \{c_n\} + \{\eta_n\} \star \{c_n\} \\ &= \{I_n\} \star (\{\delta_n\} + \{e_n\}) + \{\eta_n\} \star \{c_n\} \\ &= \{I_n\} + \{I_n\} \star \{e_n\} + \{\eta_n\} \star \{c_n\} \end{aligned} \tag{10.5–17}$$

In Equation 10.5–17 the term $\{I_n\}$ represents the desired data sequence, the term $\{I_n\} \star \{e_n\}$ represents the residual ISI, and the term $\{\eta_n\} \star \{c_n\}$ represents the additive noise. Our problem is to utilize the deconvolved sequence $\{\hat{I}_n\}$ to find the "best" estimate of a desired response, denoted in general by $\{d_n\}$. In the case of adaptive equalization using a training sequence, $\{d_n\} = \{I_n\}$. In a blind equalization mode, we shall generate a desired response from $\{\hat{I}_n\}$.

The mean square error (MSE) criterion may be employed to determine the "best" estimate of $\{I_n\}$ from the observed equalizer output $\{\hat{I}_n\}$. Since the transmitted sequence $\{I_n\}$ has a non-Gaussian PDF, the MSE estimate is a non-linear transformation of $\{\hat{I}_n\}$.

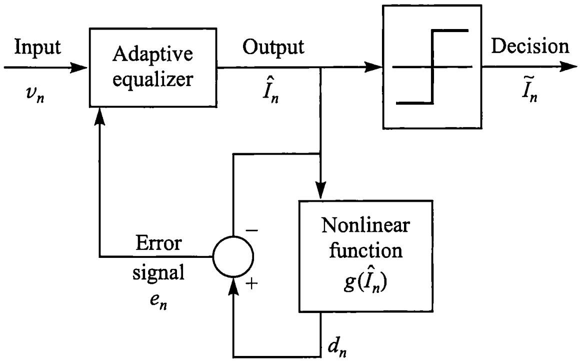


FIGURE 10.5–1
Adaptive blind equalization with stochastic gradient algorithms.

In general, the best estimate $\{d_n\}$ is given by

$$\begin{aligned} d_n &= g(\hat{I}_n) && \text{(memoryless)} \\ d_n &= g(\hat{I}_n, \hat{I}_{n-1}, \ldots, \hat{I}_{n-m}) && (m\text{th-order memory}) \end{aligned} \qquad (10.5\text{–}18)$$

where $g(\ )$ is a non-linear function. The sequence $\{d_n\}$ is then used to generate an error signal, which is fed back into the adaptive equalization filter, as shown in Figure 10.5–1. Let us consider the nonlinear function based on the MSE criterion.

A well-known classical estimation problem is the following. If the equalizer output $\hat{I}_n$ is expressed as

$$\hat{I}_n = I_n + \tilde{\eta}_n \qquad (10.5\text{–}19)$$

where $\tilde{\eta}_n$ is assumed to be zero-mean Gaussian (the central limit theorem may be invoked here for the residual ISI and the additive noise), $\{I_n\}$ and $\{\tilde{\eta}_n\}$ are statistically independent, and $\{I_n\}$ are statistically independent and identically distributed random variables, then the MSE estimate of $\{I_n\}$ is

$$d_n = E(I_n | \hat{I}_n) \qquad (10.5\text{–}20)$$

which is a non-linear function of the equalizer output when $\{I_n\}$ is non-Gaussian.

Table 10.5–1 illustrates the general form of existing blind equalization algorithms that are based on LMS adaptation. We observe that the basic difference among these algorithms lies in the choice of the memoryless non-linearity. The most widely used algorithm in practice is the *Godard algorithm*, sometimes also called the *constant-modulus algorithm* (CMA).

It is apparent from Table 10.5–1 that the output sequence $\{d_n\}$ obtained by taking a non-linear function of the equalizer output plays the role of the desired response or a training sequence. It is also apparent that these algorithms are simple to implement, since they are basically LMS-type algorithms. As such, we expect that the convergence characteristics of these algorithms will depend on the autocorrelation matrix of the received data $\{v_n\}$.

With regard to convergence, the adaptive LMS-type algorithms converge in the mean when

$$E\left[v_n g^*(\hat{I}_n)\right] = E\left[v_n \hat{I}_n^*\right] \qquad (10.5\text{–}21)$$

TABLE 10.5–1
Stochastic Gradient Algorithms for Blind Equalization

Equalizer tap coefficients	$\{c_n, 0 \le n \le N-1\}$
Received signal sequence	$\{v_n\}$
Equalizer output sequence	$\{\hat{I}_n\} = \{v_n\} \star \{c_n\}$
Equalizer error sequence	$\{e_n\} = g(\hat{I}_n) - \hat{I}_n$
Tap coefficient update equation	$\boldsymbol{c}_{n+1} = \boldsymbol{c}_n + \Delta \boldsymbol{v}_n^* e_n$

Algorithm	**Non-linearity: $g(\tilde{I}_n)$**
Godard	$\dfrac{\hat{I}_n}{\lvert\tilde{I}_n\rvert}(\lvert\tilde{I}_n\rvert + R_2\lvert\hat{I}_n\rvert - \lvert\hat{I}_n\rvert^3)$, $R_2 = \dfrac{E\{\lvert I_n\rvert^4\}}{E\{\lvert I_n\rvert^2\}}$
Sato	$\zeta\,\mathrm{csgn}(\hat{I}_n)$, $\zeta = \dfrac{E\{[\mathrm{Re}(I_n)]^2\}}{E\{\lvert\mathrm{Re}(I_n)\rvert\}}$
Benveniste–Goursat	$\hat{I}_n + k_1(\hat{I}_n - I_n) + k_2\lvert\hat{I}_n - \tilde{I}_n\rvert[\zeta\ \mathrm{csgn}\,(\hat{I}_n) - \tilde{I}_n]$, k_1 and k_2 are positive constants
Stop-and-go	$\hat{I}_n + \frac{1}{2}A(\hat{I}_n - \tilde{I}_n) + \frac{1}{2}B(\hat{I}_n - \tilde{I}_n)^*$, $(A, B) = (2, 0), (1, 1), (1, -1)$, or $(0, 0)$, depending on the signs of decision-directed error $\hat{I}_n - \tilde{I}_n$ and the error $\zeta\ \mathrm{csgn}\,(\hat{I}_n) - \tilde{I}_n$

and, in the mean square sense, when

$$\begin{aligned} E\left[\boldsymbol{C}_n^H v_n g^*(\hat{I}_n)\right] &= E\left[\boldsymbol{C}_n^H v_n \hat{I}_n^*\right] \\ E\left[\hat{I}_n g^*(\hat{I}_n)\right] &= E\left[\lvert\hat{I}_n\rvert^2\right] \end{aligned} \tag{10.5–22}$$

Therefore, it is required that the equalizer output $\{\hat{I}_n\}$ satisfy Equation 10.5–22. Note that Equation 10.5–22 states that the autocorrelation of $\{\hat{I}_n\}$ (the right-hand side) equals the cross correlation between $\hat{I}_n$ and a non-linear transformation of $\hat{I}_n$ (left-hand side). Processes that satisfy this property are called *Bussgang* (1952), as named by Bellini (1986). In summary, the algorithms given in Table 10.5–1 converge when the equalizer output sequence $\hat{I}_n$ satisfies the Bussgang property.

The basic limitation of stochastic gradient algorithms is their relatively slow convergence. Some improvement in the convergence rate can be achieved by modifying the adaptive algorithms from LMS-type to RLS-type.

Godard algorithm The Godard blind equalization algorithm is a steepest-descent algorithm that is widely used in practice when a training sequence is not available. Let us describe this algorithm in more detail, assuming a general QAM signal constellation.

Godard considered the problem of combined equalization and carrier phase recovery and tracking. The carrier phase tracking is performed at baseband, following the equalizer as shown in Figure 10.5–2. Based on this structure, we may express the equalizer output as

$$\hat{I}_k = \sum_{n=-K}^{K} c_n v_{k-n} \tag{10.5–23}$$

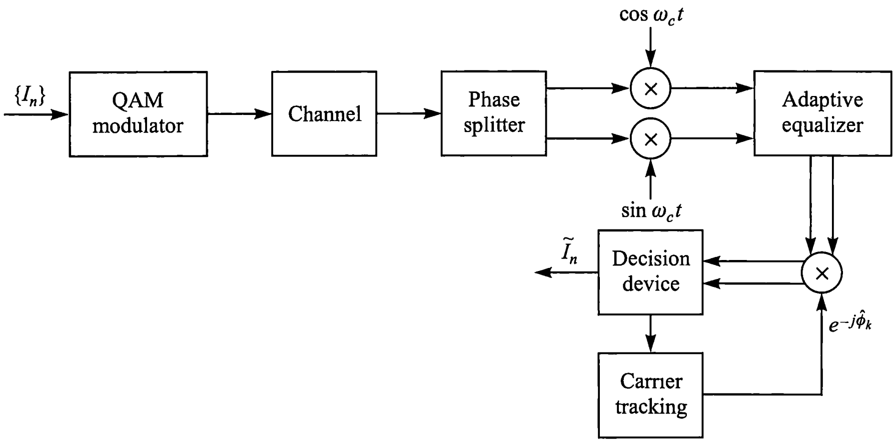


FIGURE 10.5–2
Godard scheme for combined adaptive (blind) equalization and carrier phase tracking.

and the input to the decision device as $\hat{I}_n \exp(-j\hat{\phi}_k)$, where $\hat{\phi}_k$ is the carrier phase estimate in the kth symbol interval.

If the desired symbol were known, we could form the error signal

$$\varepsilon_k = I_k - \hat{I}_k e^{-j\hat{\phi}_k} \tag{10.5–24}$$

and minimize the MSE with respect to $\hat{\phi}_k$ and $\{c_n\}$, i.e.,

$$\min_{\hat{\phi}_k, \boldsymbol{C}} E\left(|I_k - \hat{I}_k e^{-j\hat{\phi}_k}|^2\right) \tag{10.5–25}$$

This criterion leads us to use the LMS algorithm for recursively estimating $\boldsymbol{C}$ and ϕ_k. The LMS algorithm based on knowledge of the transmitted sequence is

$$\hat{\boldsymbol{C}}_{k+1} = \hat{\boldsymbol{C}}_k + \Delta_c\left(I_k - \hat{I}_k e^{-j\hat{\phi}_k}\right)\boldsymbol{V}_k^* e^{j\hat{\phi}_k} \tag{10.5–26}$$

$$\hat{\phi}_{k+1} = \hat{\phi}_k + \Delta_\phi \mathrm{Im}\left(I_k \hat{I}_k^* e^{j\hat{\phi}_k}\right) \tag{10.5–27}$$

where Δ_c and Δ_ϕ are the step-size parameters for the two recursive equations. Note that these recursive equations are coupled together. Unfortunately, these equations will not converge, in general, when the desired symbol sequence $\{I_k\}$ is unknown.

The approach proposed by Godard is to use a criterion that depends on the amount of intersymbol interference at the output of the equalizer but one that is independent of the QAM signal constellation and the carrier phase. For example, a cost function that is independent of carrier phase and has the property that its minimum leads to a small MSE is

$$G^{(p)} = E\left(|\hat{I}_k|^p - |I_k|^p\right)^2 \tag{10.5–28}$$

where p is a positive and real integer. Minimization of $G^{(p)}$ with respect to the equalizer coefficients results in the equalization of the signal amplitude only. Based on this observation, Godard selected a more general cost function, called the *dispersion of* *order p*, defined as

$$D^{(p)} = E\big(|\hat{I}_k|^p - R_p\big)^2 \tag{10.5–29}$$

where R_p is a positive real constant. As in the case of $G^{(p)}$, we observe that $D^{(p)}$ is independent of the carrier phase.

Minimization of $D^{(p)}$ with respect to the equalizer coefficients can be performed recursively according to the steepest-descent algorithm

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k - \Delta_p \frac{dD^{(p)}}{d\boldsymbol{C}_k} \tag{10.5–30}$$

where Δ_p is the step-size parameter. By differentiating $D^{(p)}$ and dropping the expectation operation, we obtain the following LMS-type algorithm for adjusting the equalizer coefficients:

$$\hat{\boldsymbol{C}}_{k+1} = \hat{\boldsymbol{C}}_k + \Delta_p \boldsymbol{V}_k^* \hat{I}_k |\hat{I}_k|^{p-2}\big(R_p - |\hat{I}_k|^p\big) \tag{10.5–31}$$

where Δ_p is the step-size parameter and the optimum choice of R_p is

$$R_p = \frac{E\big(|I_k|^{2p}\big)}{E\big(|I_k|^p\big)} \tag{10.5–32}$$

As expected, the recursion in Equation 10.5–31 for $\hat{\boldsymbol{C}}_k$ does not require knowledge of the carrier phase. Carrier phase tracking may be carried out in a decision-directed mode according to Equation 10.5–27, with $\tilde{I}_k$ substituted in place of I_k.

Of particular importance is the case $p = 2$, which leads to the relatively simple algorithm

$$\begin{aligned} \hat{\boldsymbol{C}}_{k+1} &= \hat{\boldsymbol{C}}_k + \Delta_p \boldsymbol{V}_k^* \hat{I}_k \big(R_2 - |\hat{I}_k|^2\big) \\ \hat{\phi}_{k+1} &= \hat{\phi}_k + \Delta_\phi \operatorname{Im}\big(\tilde{I}_k \hat{I}_k^* e^{j\hat{\phi}_k}\big) \end{aligned} \tag{10.5–33}$$

where $\tilde{I}_k$ is the output decision based on $\hat{I}_k$, and

$$R_2 = \frac{E\big(|I_k|^4\big)}{E\big(|I_k|^2\big)} \tag{10.5–34}$$

Convergence of the algorithm given in Equation 10.5–33 is demonstrated in the paper by Godard (1980). Initially, the equalizer coefficients are set to zero except for the center (reference) tap, which is set according to the condition

$$|c_0|^2 > \frac{E|I_k|^4}{2|x_0|^2\big[E(|I_k|^2)\big]^2} \tag{10.5–35}$$

which is sufficient, but not necessary, for convergence of the algorithm. Simulation results performed by Godard on simulated telephone channels with typical frequency-response characteristics and transmission rates of 7200–12,000 bits/s indicate that the algorithm in Equation 10.5–31 performs well and leads to convergence in 5000–20,000 iterations, depending on the signal constellation. Initially, the eye pattern was closed prior to equalization. The number of iterations required for convergence is about an order of magnitude greater than the number required to equalize the channels with a known training sequence. No apparent difficulties were encountered in using the decision-directed phase estimation algorithm in Equation 10.5–33 from the beginning of the equalizer adjustment process.

10.5–3 Blind Equalization Algorithms Based on Second- and Higher-Order Signal Statistics

It is well known that second-order statistics (autocorrelation) of the received signal sequence provide information on the magnitude of the channel characteristics, but not on the phase. However, this statement is not correct if the autocorrelation function of the received signal is periodic, as is the case for a digitally modulated signal. In such a case, it is possible to obtain a measurement of the amplitude and the phase of the channel from the received signal. This cyclostationarity property of the received signal forms the basis for a channel estimation algorithm devised by Tong et al. (1994, 1995).

It is also possible to estimate the channel response from the received signal by using higher-order statistical methods. In particular, the impulse response of a linear, discrete-time-invariant system can be obtained explicitly from cumulants of the received signal, provided that the channel input is non-Gaussian. We describe the following simple method, due to Giannakis (1987) and Giannakis and Mendel (1989) for estimation of the channel impulse response from fourth-order cumulants of the received signal sequence. For simplicity, we assume that the received signal sequence is real-valued. The fourth-order cumulant is defined as

$$\begin{aligned} c(v_k, v_{k+m}, v_{k+n}, v_{k+l}) &\equiv c_r(m, n, l) \\ &= E(v_k v_{k+m} v_{k+n} v_{k+l}) \\ &\quad - E(v_k\, v_{k+m}) E(v_{k+n} v_{k+l}) \\ &\quad - E(v_k\, v_{k+n}) E(v_{k+m} v_{k+l}) \\ &\quad - E(v_k\, v_{k+l}) E(v_{k+m} v_{k+n}) \end{aligned} \tag{10.5–36}$$

(The fourth-order cumulant of a Gaussian signal process is zero.) Consequently, it follows that

$$c_r(m, n, l) = c(I_k, I_{k+m}, I_{k+n}, I_{k+l}) \sum_{k=0}^{\infty} f_k f_{k+m} f_{k+n} f_{k+l} \tag{10.5–37}$$

For a statistically independent and identically distributed input sequence $\{I_n\}$ to the channel, $c(I_k, I_{k+m}, I_{k+n}, I_{k+l}) = k$, a constant, which is called the *kurtosis*. Then, if the length of the channel response is $L + 1$, we may let $m = n = l = -L$ so that

$$c_r(-L, -L, -L) = k f_L f_0^3 \tag{10.5–38}$$

Similarly, if we let $m = 0$, $n = L$, and $l = p$, we obtain

$$c_r(0, L, p) = k f_L f_0^2 f_p \tag{10.5–39}$$

If we combine Equations 10.5–38 and 10.5–39, we obtain the impulse response within a scale factor as

$$f_p = f_0 \frac{c_r(0, L, p)}{c_r(-L, -L, -L)}, \qquad p = 1, 2, \ldots, L \tag{10.5–40}$$

The cumulants $c_r(m, n, l)$ are estimated from sample averages of the received signal sequence $\{v_n\}$.

Another approach based on higher-order statistics is due to Hatzinakos and Nikias (1991). They have introduced the first polyspectra-based adaptive blind equalization method named the *tricepstrum equalization algorithm* (TEA). This method estimates the channel response characteristics by using the complex cepstrum of the fourth-order cumulants (tricepstrum) of the received signal sequence $\{v_n\}$. TEA depends only on fourth-order cumulants of $\{v_n\}$ and is capable of separately reconstructing the minimum-phase and maximum-phase characteristics of the channel. The channel equalizer coefficients are then computed from the measured channel characteristics. The basic approach used in TEA is to compute the tricepstrum of the received sequence $\{v_n\}$, which is the inverse (three-dimensional) Fourier transform of the logarithm of the trispectrum of $\{v_n\}$. [The *trispectrum* is the three-dimensional discrete Fourier transform of the fourth-order cumulant sequence $c_r(m, n, l)$.] The equalizer coefficients are then computed from the cepstral coefficients.

By separating the channel estimation from the channel equalization, it is possible to use any type of equalizer for the ISI, i.e., either linear, or decision-feedback, or maximum-likelihood sequence detection. The major disadvantage with this class of algorithms is the large amount of data and the inherent computational complexity involved in the estimation of the higher-order moments (cumulants) of the received signal.

In conclusion, we have provided an overview of three classes of blind equalization algorithms that find applications in digital communications. Of the three families of algorithms described, those based on the maximum-likelihood criterion for jointly estimating the channel impulse response and the data sequence are optimal and require relatively few received signal samples for performing channel estimation. However, the computational complexity of the algorithms is large when the ISI spans many symbols. On some channels, such as the mobile radio channel, where the span of the ISI is relatively short, these algorithms are simple to implement. However, on telephone channels, where the ISI spans many symbols but is usually not too severe, the LMS-type (stochastic gradient) algorithms are generally employed.

10.6
BIBLIOGRAPHICAL NOTES AND REFERENCES

Adaptive equalization for digital communications was developed by Lucky (1965, 1966). His algorithm was based on the peak distortion criterion and led to the zero-forcing algorithm. Lucky's work was a major breakthrough, which led to the rapid development of high-speed modems within 5 years of publication of his work. Concurrently, the LMS algorithm was devised by Widrow (1966, 1970), and its use for adaptive equalization for two-dimensional (in-phase and quadrature components) signals was described and analyzed in a tutorial paper by Proakis and Miller (1969).

A tutorial treatment of adaptive equalization algorithms that were developed during the period 1965–1975 is given by Proakis (1975). A more recent tutorial treatment of adaptive equalization is given in the paper by Qureshi (1985). The major breakthrough in adaptive equalization techniques, beginning with the work of Lucky in 1965 coupled with the development of trellis-coded modulation, which was described by Ungerboeck and Csajka (1976), has led to the development of commercially available high-speed modems with a capability of speeds exceeding 30,000 bits/s on telephone channels.

The use of a more rapidly converging algorithm for adaptive equalization was proposed by Godard (1974). Our derivation of the RLS (Kalman) algorithm, described in Section 10.4–1, follows the approach outlined by Picinbono (1978). RLS lattice algorithms for general signal estimation applications were developed by Morf (1977), Morf and Lee (1978), and Morf et al. (1977a,b,c). The applications of these algorithms have been investigated by several researchers, including Makhoul (1978), Satorius and Pack (1981), Satorius and Alexander (1979), and Ling and Proakis (1982, 1984a–c, 1985, 1986). The fast RLS Kalman algorithm for adaptive equalization was first described by Falconer and Ljung (1978). The above references are just a few of the important papers that have been published on RLS algorithms for adaptive equalization and other applications. A comprehensive treatment of RLS algorithms is given in the books by Haykin (2002) and Proakis et al. (2002).

Sato's (1975) original work on blind equalization was focused on PAM (one-dimensional) signal constellations. Subsequently it was generalized to two-dimensional and multidimensional signal constellations in the algorithms devised by Godard (1980), Benveniste and Goursat (1984), Sato et al. (1986), Foschini (1985), Picchi and Prati (1987), and Shalvi and Weinstein (1990). Blind equalization methods based on the use of second- and higher-order moments of the received signal were proposed by Giannakis (1987), Giannakis and Mendel (1989), Hatzinakos and Nikias (1991), and Tong et al. (1994, 1995). The use of the maximum-likelihood criterion for joint channel estimation and data detection has been investigated and treated in papers by Sato (1994), Seshadri (1994), Ghosh and Weber (1991), Zervas et al. (1991), and Raheli et al. (1995). Finally, the convergence characteristics of stochastic gradient blind equalization algorithms have been investigated by Ding (1990), Ding et al. (1989), and Johnson (1991).

PROBLEMS

10.1 An equivalent discrete-time channel with white Gaussian noise is shown in Figure P10.1

a. Suppose we use a linear equalizer to equalize the channel. Determine the tap coefficients c_{-1}, c_0, c_1 of a three-tap equalizer. To simplify the computation, let the AWGN be zero.

b. The tap coefficients of the linear equalizer in (a) are determined recursively via the algorithm

$$\boldsymbol{C}_{k+1} = \boldsymbol{C}_k - \Delta \boldsymbol{G}_k, \qquad \boldsymbol{C}_k = [c_{-1k} \quad c_{0k} \quad c_{1k}]^t$$

where $\boldsymbol{G}_k = \boldsymbol{\Gamma}\boldsymbol{C}_k - \boldsymbol{\xi}$ is the gradient vector and Δ is the step size. Determine the range of values of Δ to ensure convergence of the recursive algorithm. To simplify the computation, let the AWGN be zero.

c. Determine the tap weights of a DFE with two feedforward taps and one feedback tap. To simplify the computation, let the AWGN be zero.

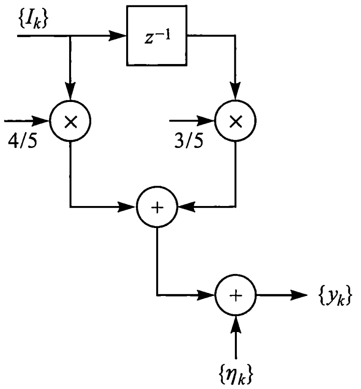


FIGURE P10.1

10.2 Refer to Problem 9.49 and answer the following questions.

a. Determine the maximum value of Δ that can be used to ensure that the equalizer coefficients converge during operation in the adaptive mode.

b. What is the variance of the self-noise generated by the three-tap equalizer when operating in an adaptive mode, as a function of Δ? Suppose it is desired to limit the variance of the self-noise to 10 percent of the minimum MSE for the three-tap equalizer when $N_0 = 0.1$. What value of Δ would you select?

c. If the optimum coefficients of the equalizer are computed recursively by the method of steepest descent, the recursive equation can be expressed in the form

$$\boldsymbol{C}_{n+1} = (\boldsymbol{I} - \Delta\boldsymbol{\Gamma})\boldsymbol{C}_n + \Delta\boldsymbol{\xi}$$

where $\boldsymbol{I}$ is the identity matrix. The above represents a set of three coupled first-order difference equations. They can be decoupled by a linear transformation that diagonalizes the matrix $\boldsymbol{\Gamma}$. That is, $\boldsymbol{\Gamma} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^t$ where $\boldsymbol{\Lambda}$ is the diagonal matrix having the eigenvalues of $\boldsymbol{\Gamma}$ as its diagonal elements and $\boldsymbol{U}$ is the (normalized) modal matrix that can be obtained from your answer to Problem 9.49(b). Let $\boldsymbol{C}' = \boldsymbol{U}^t\boldsymbol{C}$ and determine the steady-state solution for $\boldsymbol{C}'$. From this, evaluate $\boldsymbol{C} = (\boldsymbol{U}^t)^{-1}\boldsymbol{C}' = \boldsymbol{U}\boldsymbol{C}'$ and, thus, show that your answer agrees with the result obtained in Problem 9.49(a).

10.3 When a periodic pseudorandom sequence of length N is used to adjust the coefficients of an N-tap linear equalizer, the computations can be performed efficiently in the frequency domain by use of the discrete Fourier transform (DFT). Suppose that $\{y_n\}$ is a sequence of N received samples (taken at the symbol rate) at the equalizer input. Then the computation of the equalizer coefficients is performed as follows.

a. Compute the DFT of one period of the equalizer input sequence $\{y_n\}$, i.e.,

$$Y_k = \sum_{n=0}^{N-1} y_n e^{-j2\pi nk/N}$$

b. Compute the desired equalizer spectrum

$$C_k = \frac{X_k Y_k^*}{|Y_k|^2}, \qquad k = 0, 1, \ldots, N-1$$

where $\{X_i\}$ is the precomputed DFT of the training sequence.

c. Compute the inverse DFT of $\{C_k\}$ to obtain the equalizer coefficients $\{c_n\}$. Show that this procedure in the absence of noise yields an equalizer whose frequency response is equal to the frequency response of the inverse folded channel spectrum at the N uniformly spaced frequencies $f_k = k/NT$, $k = 0, 1, \ldots, N-1$.

10.4 Show that the gradient vector in the minimization of the MSE may be expressed as

$$\boldsymbol{G}_k = -E(\varepsilon_k \boldsymbol{V}_k^*)$$

where the error $\varepsilon_k = I_k - \hat{I}_k$, and the estimate of $\boldsymbol{G}_k$, i.e.,

$$\hat{\boldsymbol{G}}_k = -\varepsilon_k \boldsymbol{V}_k^*$$

satisfies the condition that $E(\hat{\boldsymbol{G}}_k) = \boldsymbol{G}_k$.

10.5 The *tap-leakage LMS algorithm* proposed in the paper by Gitlin et al. (1982) may be expressed as

$$\boldsymbol{C}_N(n+1) = w\boldsymbol{C}_N(n) + \Delta\varepsilon(n)\boldsymbol{V}_N^*(n)$$

where $0 < w < 1$, Δ is the step size, and $\boldsymbol{V}_N(n)$ is the data vector at time n. Determine the condition for the convergence of the mean value of $\boldsymbol{C}_N(n)$.

10.6 Consider the random process

$$x(n) = gv(n) + w(n), \qquad n = 0, 1, \ldots, M-1$$

where $v(n)$ is a known sequence, g is a random variable with $E(g) = 0$, and $E(g^2) = G$. The process $w(n)$ is a white noise sequence with

$$\gamma_{ww}(m) = \sigma_w^2 \delta_m$$

Determine the coefficients of the linear estimator for g, that is,

$$\hat{g} = \sum_{n=0}^{M-1} h(n)x(n)$$

that minimize the mean square error.

10.7 A digital transversal filter can be realized in the frequency-sampling form with system function (see Problem 9.56)

$$\begin{aligned} H(z) &= \frac{1 - z^{-M}}{M} \sum_{k=0}^{M-1} \frac{H_k}{1 - e^{j2\pi k/M} z^{-1}} \\ &= H_1(z)H_2(z) \end{aligned}$$

where $H_1(z)$ is the comb filter, $H_2(z)$ is the parallel bank of resonators, and $\{H_k\}$ are the values of the discrete Fourier transform (DFT).

a. Suppose that this structure is implemented as an adaptive filter using the LMS algorithm to adjust the filter (DFT) parameters $\{H_k\}$. Give the time-update equation for these parameters. Sketch the adaptive filter structure.

b. Suppose that this structure is used as an adaptive channel equalizer in which the desired signal is

$$d(n) = \sum_{k=0}^{M-1} A_k \cos \omega_k n, \qquad \omega_k = \frac{2\pi k}{M}$$

With this form for the desired signal, what advantages are there in the LMS adaptive algorithm for the DFT coefficients $\{H_k\}$ over the direct-form structure with coefficients $\{h(n)\}$? [See Proakis (1970).]

10.8 Consider the performance index

$$J = h^2 + 40h + 28$$

Suppose that we search for the minimum of J by using the steepest-descent algorithm

$$h(n+1) = h(n) - \tfrac{1}{2}\Delta g(n)$$

where $g(n)$ is the gradient.

a. Determine the range of values of Δ that provides an overdamped system for the adjustment process.

b. Plot the expression for J as a function of n for a value of Δ in this range.

10.9 Determine the coefficients a_1 and a_2 for the linear predictor shown in Figure P10.9, given that the autocorrelation $\gamma_{xx}(m)$ of the input signal is

$$\gamma_{xx}(m) = b^{|m|}, \qquad 0 < b < 1$$

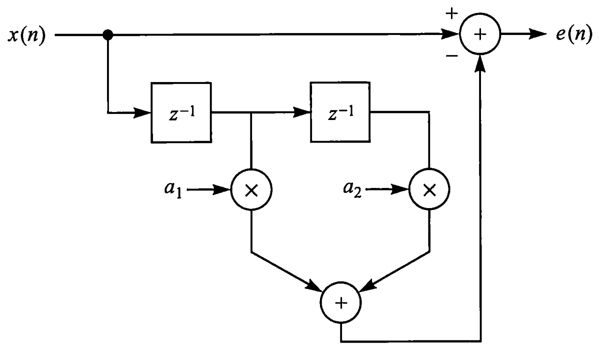


FIGURE P10.9

10.10 Determine the lattice filter and its optimum reflection coefficients corresponding to the linear predictor in Problem 10.9.

10.11 Consider the adaptive FIR filter shown in Figure P10.11. The system $C(z)$ is characterized by the system function

$$C(z) = \frac{1}{1 - 0.9z^{-1}}$$

Determine the optimum coefficients of the adaptive transversal (FIR) filter $B(z) = b_0 + b_1 z^{-1}$ that minimize the mean square error. The additive noise is white with variance $\sigma_w^2 = 0.1$.

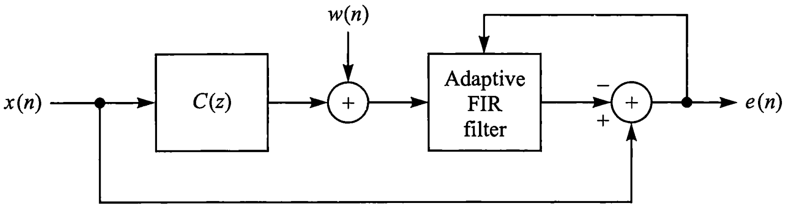


FIGURE P10.11

10.12 An $N \times N$ correlation matrix $\boldsymbol{\Gamma}$ has eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_N > 0$ and associated eigenvectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N$. Such a matrix can be represented as

$$\boldsymbol{\Gamma} = \sum_{i=1}^{N} \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^H$$

a. If $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}^{1/2}\boldsymbol{\Gamma}^{1/2}$, where $\boldsymbol{\Gamma}^{1/2}$ is the square root of $\boldsymbol{\Gamma}$, show that $\boldsymbol{\Gamma}^{1/2}$ can be represented as

$$\boldsymbol{\Gamma}^{1/2} = \sum_{i=1}^{N} \lambda_i^{1/2} \boldsymbol{v}_i \boldsymbol{v}_i^H$$

b. Using this representation, determine a procedure for computing $\boldsymbol{\Gamma}^{1/2}$.

11

Multichannel and Multicarrier Systems

In some applications, it is desirable to transmit the same information-bearing signal over several channels. This mode of transmission is used primarily in situations where there is a high probability that one or more of the channels will be unreliable from time to time. For example, radio channels such as ionospheric scatter and tropospheric scatter suffer from signal fading due to multipath, which renders the channels unreliable for short periods of time. As another example, multichannel signaling is sometimes employed in wireless communication systems as a means of overcoming the effects of interference of the transmitted signal. By transmitting the same information over multiple channels, we are providing signal diversity, which the receiver can exploit to recover the information.

Another form of multichannel communications is multiple carrier transmission, where the frequency band of the channel is subdivided into a number of subchannels and information is transmitted on each of the subchannels. A rationale for subdividing the frequency band of a channel into a number of narrowband channels is given below.

In this chapter, we consider both multichannel signal transmission and multicarrier transmission. The focus is on the performance of such systems in AWGN channels. The performance of multichannel and multicarrier transmission in fading channels is treated in Chapter 13. We begin with a treatment of multichannel transmission.

11.1
MULTICHANNEL DIGITAL COMMUNICATIONS IN AWGN CHANNELS

In this section, we confine our attention to multichannel signaling over fixed channels that differ only in attenuation and phase shift. The specific model for the multichannel digital signaling system is illustrated in Figure 11.1–1 and may be described as follows. The signal waveforms, in general, are expressed as

$$s_m^{(n)}(t) = \mathrm{Re}\left[s_{lm}^{(n)}(t)e^{j2\pi f_c t}\right], \qquad 0 \le t \le T$$
$$n = 1, 2, \ldots, L, \qquad m = 1, 2, \ldots, M \quad (11.1\text{–}1)$$

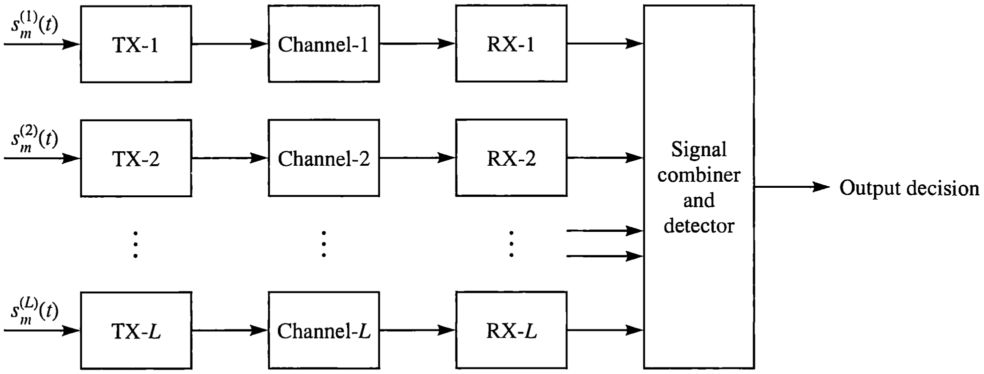


FIGURE 11.1-1
Model of a multichannel digital communication system.

where L is the number of channels and M is the number of waveforms. The waveforms are assumed to have equal energy and to be equally probable a priori. The waveforms $\{s_m^{(n)}(t)\}$ transmitted over the L channels are scaled by the attenuation factors $\{\alpha_n\}$, phase-shifted by $\{\phi_n\}$, and corrupted by additive noise. The equivalent low-pass signals received from the L channels may be expressed as

$$r_l^{(n)}(t) = \alpha_n e^{j\phi_n} s_{lm}^{(n)}(t) + z_n(t), \quad 0 \le t \le T$$
$$n = 1, 2, \ldots, L, \quad m = 1, 2, \ldots, M \qquad (11.1\text{–}2)$$

where $\{s_{lm}^{(n)}(t)\}$ are the equivalent lowpass transmitted waveforms and $\{z_n(t)\}$ represent the additive noise processes on the L channels. We assume that $\{z_n(t)\}$ are mutually statistically independent and identically distributed Gaussian noise random processes.

We consider two types of processing at the receiver, namely, coherent detection and noncoherent detection. The receiver for coherent detection estimates the channel parameters $\{\alpha_n\}$ and $\{\phi_n\}$ and uses the estimates in computing the decision variables. Suppose we define $g_n = \alpha_n e^{j\phi_n}$ and let $\hat{g}_n$ be the estimate of g_n. The multichannel receiver correlates each of the L received signals with a replica of the corresponding transmitted signals, multiplies each of the correlator outputs by the corresponding estimates $\{\hat{g}_n^*\}$, and sums the resulting signals. Thus, the decision variables for coherent detection are the correlation metrics

$$CM_m = \sum_{n=1}^{L} \mathrm{Re}\left[\hat{g}_n^* \int_0^T r_l^{(n)}(t) s_{lm}^{(n)*}(t)\,dt\right], \qquad m = 1, 2, \ldots, M \qquad (11.1\text{–}3)$$

In noncoherent detection, no attempt is made to estimate the channel parameters. The demodulator may base its decision either on the sum of the envelopes (envelope detection) or the sum of the squared envelopes (square-law detection) of the matched filter outputs. In general, the performance obtained with envelope detection differs little from the performance obtained with square-law detection in AWGN. However, square-law detection of multichannel signaling in AWGN channels is considerably easier to analyze than envelope detection. Therefore, we confine our attention to square-law detection of the received signals of the L channels, which produces the decision variables

$$CM_m = \sum_{n=1}^{L} \left| \int_0^T r_l^{(n)}(t) s_{lm}^{(n)*}(t)\, dt \right|^2, \qquad m = 1, 2, \ldots, M \tag{11.1–4}$$

Let us consider binary signaling first, and assume that $s_{l1}^{(n)}(t)$, $n = 1, 2, \ldots, L$, are the L transmitted waveforms. Then an error is committed if $CM_2 > CM_1$, or, equivalently, if the difference $D = CM_1 - CM_2 < 0$. For noncoherent detection, this difference may be expressed as

$$D = \sum_{n=1}^{L} \left(|X_n|^2 - |Y_n|^2 \right) \tag{11.1–5}$$

where the variables $\{X_n\}$ and $\{Y_n\}$ are defined as

$$\begin{aligned} X_n &= \int_0^T r_l^{(n)}(t) s_{l1}^{(n)*}(t)\, dt, \qquad n = 1, 2, \ldots, L \\ Y_n &= \int_0^T r_l^{(n)}(t) s_{l2}^{(n)*}(t)\, dt, \qquad n = 1, 2, \ldots, L \end{aligned} \tag{11.1–6}$$

The $\{X_n\}$ are mutually independent and identically distributed complex Gaussian random variables. The same statement applies to the variables $\{Y_n\}$. However, for any n, X_n and Y_n may be correlated. For coherent detection, the difference $D = CM_1 - CM_2$ may be expressed as

$$D = \tfrac{1}{2} \sum_{n=1}^{L} \left(X_n Y_n^* + X_n^* Y_n \right) \tag{11.1–7}$$

where, by definition,

$$\begin{aligned} Y_n &= \hat{g}_n, \qquad n = 1, 2, \ldots, L \\ X_n &= \int_0^T r_l^{(n)}(t) \left[s_{l1}^{(n)*}(t) - s_{l2}^{(n)*}(t) \right] dt \end{aligned} \tag{11.1–8}$$

If the estimates $\{\hat{g}_n\}$ are obtained from observation of the received signal over one or more signaling intervals, as described in Appendix C, their statistical characteristics are described by the Gaussian distribution. Then the $\{Y_n\}$ are characterized as mutually independent and identically distributed Gaussian random variables. The same statement applies to the variables $\{X_n\}$. As in noncoherent detection, we allow for correlation between X_n and Y_n, but not between X_m and Y_n for $m \neq n$.

11.1–1 Binary Signals

In Appendix B, we derive the probability that the general quadratic form

$$D = \sum_{n=1}^{L} \left(A|X_n|^2 + B|Y_n|^2 + C X_n Y_n^* + C^* X_n^* Y_n \right) \tag{11.1–9}$$

in complex-valued Gaussian random variables is less than zero, where A and B are real constants and C may be either a real or a complex-valued constant. This probability, which is given in Equation B–21 of Appendix B, is the probability of error for binary multichannel signaling in AWGN. A number of special cases are of particular importance.

If the binary signals are antipodal and the estimates of $\{g_n\}$ are perfect, as in coherent PSK, the probability of error takes the simple form

$$P_b = Q(\sqrt{2\gamma_b}) \tag{11.1–10}$$

where

$$\gamma_b = \frac{\mathcal{E}}{N_0}\sum_{n=1}^{L}|g_n|^2 = \frac{\mathcal{E}}{N_0}\sum_{n=1}^{L}\alpha_n^2 \tag{11.1–11}$$

is the SNR per bit. If the channels are all identical, $\alpha_n = \alpha$ for all n and, hence,

$$\gamma_b = \frac{L\mathcal{E}}{N_0}\alpha^2 \tag{11.1–12}$$

We observe that $L\mathcal{E}$ is the total transmitted signal energy for the L signals. The interpretation of this result is that the receiver combines the energy from the L channels in an optimum manner. That is, there is no loss in performance in dividing the total transmitted signal energy among the L channels. The same performance is obtained as in the case in which a single waveform having energy $L\mathcal{E}$ is transmitted on one channel. This behavior holds true only if the estimates $\hat{g}_n = g_n$, for all n. If the estimates are not perfect, a loss in performance occurs, the amount of which depends on the quality of the estimates, as described in Appendix C.

Perfect estimates for $\{g_n\}$ constitute an extreme case. At the other extreme, we have binary DPSK signaling. In DPSK, the estimates $\{\hat{g}_n\}$ are simply the (normalized) signal-plus-noise samples at the outputs of the matched filters in the previous signaling interval. This is the simplest estimate that one might consider using in estimating $\{g_n\}$. For binary DPSK, the probability of error obtained from Equation B–21 is

$$P_b = \frac{1}{2^{2L-1}}e^{-\gamma_b}\sum_{n=0}^{L-1}c_n\gamma_b^n \tag{11.1–13}$$

where, by definition,

$$c_n = \frac{1}{n!}\sum_{k=0}^{L-1-n}\binom{2L-1}{k} \tag{11.1–14}$$

and γ_b is the SNR per bit defined in Equation 11.1–11 and, for identical channels, in Equation 11.1–12. This result can be compared with the single-channel ($L = 1$) error probability. To simplify the comparison, we assume that the L channels have identical attenuation factors. Thus, for the same value of γ_b, the performance of the multichannel system is poorer than that of the single-channel system. That is, splitting the total transmitted energy among L channels results in a loss in performance, the amount of which depends on L.

A loss in performance also occurs in square-law detection of orthogonal signals transmitted over L channels. For binary orthogonal signaling, the expression for the probability of error is identical in form to that for binary DPSK given in Equation 11.1–13, except that γ_b is replaced by $\frac{1}{2}\gamma_b$. That is, binary orthogonal signaling with noncoherent detection is 3 dB poorer than binary DPSK. However, the loss in performance due to noncoherent combination of the signals received on the L channels is identical to that for binary DPSK.

Figure 11.1–2 illustrates the loss resulting from noncoherent (square-law) combining of the L signals as a function of L. The probability of error is not shown, but it can be easily obtained from the curve of the expression

$$P_b = \tfrac{1}{2}e^{-\gamma_b} \tag{11.1–15}$$

which is the error probability of binary DPSK shown in Figure 4.5–5 and then degrading the required SNR per bit, γ_b, by the noncoherent combining loss corresponding to the value of L.

11.1–2 *M*-ary Orthogonal Signals

Now let us consider M-ary orthogonal signaling with square-law detection and combination of the signals on the L channels. The decision variables are given by Equation 11.1–4. Suppose that the signals $s_{l1}^{(n)}(t)$, $n = 1, 2, \ldots, L$, are transmitted over the

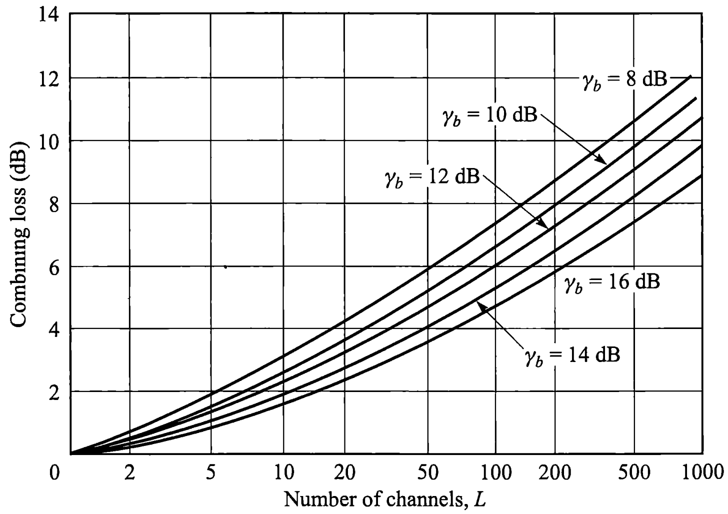


FIGURE 11.1–2
Combining loss in noncoherent detection and combination of binary multichannel signals.

L AWGN channels. Then, the decision variables are expressed as

$$\begin{aligned} CM_1 &\equiv U_1 = \sum_{n=1}^{L} |2\mathcal{E}\alpha_n + N_{n1}|^2 \\ CM_m &\equiv U_m = \sum_{n=1}^{L} |N_{nm}|^2, \qquad m = 2, 3, \ldots, M \end{aligned} \tag{11.1–16}$$

where the $\{N_{nm}\}$ are circular complex-valued zero-mean Gaussian random variables with variance $\sigma^2 = 2\mathcal{E}N_0$ per real and imaginary component. Hence U_1 is described statistically as a noncentral chi-square random variable with $2L$ degrees of freedom and noncentrality parameter

$$s^2 = \sum_{n=1}^{L} (2\mathcal{E}\alpha_n)^2 = 4\mathcal{E}^2 \sum_{n=1}^{L} \alpha_n^2 \tag{11.1–17}$$

Using Equation 2.3–29, we obtain the PDF of U_1 as

$$p(u_1) = \frac{1}{4\mathcal{E}N_0}\left(\frac{u_1}{s^2}\right)^{(L-1)/2} \exp\left(-\frac{s^2 + u_1}{4\mathcal{E}N_0}\right) I_{L-1}\left(\frac{s\sqrt{u_1}}{2\mathcal{E}N_0}\right), \quad u_1 \geq 0 \tag{11.1–18}$$

On the other hand, the $\{U_m\}$, $m = 2, 3, \ldots, M$, are statistically independent and identically chi-square-distributed random variables, each having $2L$ degrees of freedom. Using Equation 2.3–21, we obtain the PDF for U_m as

$$p(u_m) = \frac{1}{(4\mathcal{E}N_0)^L (L-1)!} u_m^{L-1} e^{-u_m/4\mathcal{E}N_0}, \qquad u_m \geq 0, \quad m = 2, 3, \ldots, M \tag{11.1–19}$$

The probability of a symbol error is

$$\begin{aligned} P_e &= 1 - P_c \\ &= 1 - P(U_2 < U_1, U_3 < U_1, \ldots, U_M < U_1) \\ &= 1 - \int_0^\infty [P(U_2 < u_1 | U_1 = u_1)]^{M-1} p(u_1)\, du_1 \end{aligned} \tag{11.1–20}$$

But

$$P(U_2 < u_1 | U_1 = u_1) = 1 - \exp\left(-\frac{u_1}{4\mathcal{E}N_0}\right) \sum_{k=0}^{L-1} \frac{1}{k!}\left(\frac{u_1}{4\mathcal{E}N_0}\right)^k \tag{11.1–21}$$

Hence,

$$\begin{aligned} P_e &= 1 - \int_0^\infty \left[1 - e^{-u_1/4\mathcal{E}N_0} \sum_{k=0}^{L-1} \frac{1}{k!}\left(\frac{u_1}{4\mathcal{E}N_0}\right)^k\right]^{M-1} p(u_1)\, du_1 \\ &= 1 - \int_0^\infty \left(1 - e^{-v} \sum_{k=0}^{L-1} \frac{v^k}{k!}\right)^{M-1} \left(\frac{v}{\gamma}\right)^{(L-1)/2} e^{-(\gamma+v)} I_{L-1}(2\sqrt{\gamma v})\, dv \end{aligned} \tag{11.1–22}$$

where

$$\gamma = \mathcal{E} \sum_{n=1}^{L} \frac{\alpha_n^2}{N_0}$$

The integral in Equation 11.1–22 can be evaluated numerically. It is also possible to expand the term $(1 - x)^{M-1}$ in Equation 11.1–22 and carry out the integration term by term. This approach yields an expression for P_e in terms of finite sums.

An alternative approach is to use the union bound

$$P_e < (M - 1)P_2(L) \tag{11.1–23}$$

where $P_2(L)$ is the probability of error in choosing between U_1 and any one of the $M - 1$ decision variables $\{U_m\}$, $m = 2, 3, \ldots, M$. From our previous discussion on the performance of binary orthogonal signaling, we have

$$P_2(L) = \frac{1}{2^{2L-1}} e^{-k\gamma_b/2} \sum_{n=0}^{L-1} c_n \left(\tfrac{1}{2} k\gamma_b\right)^n \tag{11.1–24}$$

where c_n is given by Equation 11.1–14. For relatively small values of M, the union bound in Equation 11.1–23 is sufficiently tight for most practical applications.

11.2
MULTICARRIER COMMUNICATIONS

From our treatment of nonideal linear filter channels in Chapters 9 and 10, we have observed that such channels introduce ISI, which degrades performance compared with the ideal channel. The degree of performance degradation depends on the frequency-response characteristics. Furthermore, the complexity of the receiver increases as the span of the ISI increases.

In this section, we consider the transmission of information on multiple carriers contained within the allocated channel bandwidth. The primary motivation for transmitting the data on multiple carriers is to reduce ISI and, thus, eliminate the performance degradation that is incurred in single carrier modulation.

11.2–1 Single-Carrier Versus Multicarrier Modulation

Given a particular channel characteristic, the communication system designer must decide how to efficiently utilize the available channel bandwidth in order to transmit the information reliably within the transmitter power constraint and receiver complexity constraints. For a nonideal linear filter channel, one option is to employ a single-carrier system in which the information sequence is transmitted serially at some specified rate R symbols/s. In such a channel, the time dispersion is generally much greater than the reciprocal of the symbol rate, and, hence, ISI results from the nonideal frequency-response characteristics of the channel. As we have observed, an equalizer is necessary to compensate for the channel distortion.

As an example of such an approach, we cite the modems designed to transmit data through voice-band channels in the switched telephone network, which are based on the International Telecommunications Union (ITU) standard V.34. Such modems employ QAM impressed on a single carrier that is selected along with the symbol rate from a small set of specified values to obtain the maximum throughout at the desired level of performance (error rate). The channel frequency-response characteristics are measured upon initial setup of the telephone circuit, and the symbol rate and carrier frequency are selected based on this measurement.

An alternative approach to the design of a bandwidth-efficient communication system in the presence of channel distortion is to subdivide the available channel bandwidth into a number of subchannels, such that each subchannel is nearly ideal. To elaborate, suppose that $C(f)$ is the frequency response of a nonideal, band-limited channel with a bandwidth W, and that the power spectral density of the additive Gaussian noise is $\mathcal{S}_{nn}(f)$. Then we divide the bandwidth W into $N = W/\Delta f$ subbands of width Δf, where Δf is chosen sufficiently small that $|C(f)|^2/\mathcal{S}_{nn}(f)$ is approximately a constant within each subband. Furthermore, we select the transmitted signal power to be distributed in frequency as $P(f)$, subject to the constraint that

$$\int_W P(f)\,df \le P_{av} \tag{11.2–1}$$

where P_{av} is the available average power of the transmitter. Then we transmit the data on these N subchannels. Before proceeding further with this approach, we evaluate the capacity of the nonideal additive Gaussian noise channel.

11.2–2 Capacity of a Nonideal Linear Filter Channel

Recall that the capacity of an ideal, band-limited, AWGN channel is

$$C = W \log_2 \left(1 + \frac{P_{av}}{WN_0}\right) \tag{11.2–2}$$

where C is the capacity in bits/s, W is the channel bandwidth, and P_{av} is the average transmitted power. In a multicarrier system, with Δf sufficiently small the subchannel has capacity

$$C_i = \Delta f \log_2 \left[1 + \frac{\Delta f\, P(f_i)|C(f_i)|^2}{\Delta f\, \mathcal{S}_{nn}(f_i)}\right] \tag{11.2–3}$$

Hence, the total capacity of the channel is

$$C = \sum_{i=1}^{N} C_i = \Delta f \sum_{i=1}^{N} \log_2 \left[1 + \frac{P(f_i)|C(f_i)|^2}{\mathcal{S}_{nn}(f_i)}\right] \tag{11.2–4}$$

In the limit as $\Delta f \longrightarrow 0$, we obtain the capacity of the overall channel in bits/s as

$$C = \int_W \log_2 \left[1 + \frac{P(f)|C(f)|^2}{\mathcal{S}_{nn}(f_i)} \right] df \tag{11.2–5}$$

Under the constraint on $P(f)$ given by Equation 11.2–1, the choice of $P(f)$ that maximizes C may be determined by maximizing the integral

$$\int_W \left\{ \log_2 \left[1 + \frac{P(f)|C(f)|^2}{\mathcal{S}_{nn}(f)} \right] + \lambda P(f) \right\} df \tag{11.2–6}$$

where λ is a Lagrange multiplier, which is chosen to satisfy the constraint. By using the calculus of variations to perform the maximization, we find that the optimum distribution of transmitted signal power is obtained from the solution to the equation

$$\frac{1}{P(f) + \mathcal{S}_{nn}(f)/|C(f)|^2} + \lambda = 0 \tag{11.2–7}$$

Therefore, $P(f) + \mathcal{S}_{nn}(f)|C(f)|^2$ must be a constant, whose value is adjusted to satisfy the average power constraint in Equation 11.2–1. That is,

$$P(f) = \begin{cases} K - \mathcal{S}_{nn}(f)/|C(f)|^2 & f \in W \\ 0 & f \notin W \end{cases} \tag{11.2–8}$$

This expression for the channel capacity of a nonideal linear filter channel with additive Gaussian noise is due to Holsinger (1964). The basic interpretation of this result is that the signal power should be high when the channel SNR $|C(f)|^2/\mathcal{S}_{nn}(f)$ is high, and low when the channel SNR is low. This result on the transmitted power distribution is illustrated in Figure 11.2–1. Observe that if $\mathcal{S}_{nn}(f)/|C(f)|^2$ is interpreted as the bottom of a bowl of unit depth, and we pour an amount of water equal to P_{av} into the bowl, the water will distribute itself in the bowl so as to achieve capacity. This is called the *water-filling interpretation* of the optimum power distribution as a function of frequency.

It is interesting to note that the channel capacity is smallest when the channel SNR $|C(f)|^2/\mathcal{S}_{nn}(f)$ is a constant for all $f \in W$. In this case, $P(f)$ is a constant for all $f \in W$. Equivalently, if the channel frequency response is ideal, i.e., $C(f) = 1$ for $f \in W$, then the worst Gaussian noise power distribution, from the viewpoint of maximizing capacity, is white Gaussian noise.

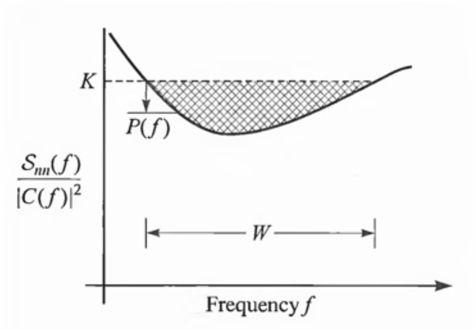


FIGURE 11.2–1
The optimum power distribution based on water-filling interpretation.

11.2–3 Orthogonal Frequency Division Multiplexing (OFDM)

The above development suggests that multicarrier modulation that divides the available channel bandwidth into subbands of relatively narrow width $\Delta f = W/N$ provides a solution that could yield transmission rates close to channel capacity. The signal in each subband may be independently coded and modulated at a synchronous symbol rate of $1/\Delta f$. If Δf is small enough, the channel frequency response $C(f)$ is essentially constant across each subband. Hence, the intersymbol interference is negligible. Such a subdivision of the channel bandwidth W is illustrated in Figure 11.2–2.

With each subband (or subchannel), we associate a sinusoidal carrier signal of the form

$$s_k(t) = \cos 2\pi f_k t, \qquad k = 0, 1, \ldots, N-1 \tag{11.2–9}$$

where f_k is the mid frequency in the kth subchannel. By selecting the symbol rate $1/T$ in each of the subchannels to be equal to the frequency separation Δf of the adjacent subcarriers, the subcarriers are orthogonal over the symbol interval T, independent of the relative phase relationship between subcarriers. That is,

$$\int_0^T \cos(2\pi f_k t + \phi_k)\cos(2\pi f_j t + \phi_j)\,dt = 0 \tag{11.2–10}$$

where $f_k - f_j = n/T$, $n = 1, 2, \ldots, N-1$, independent of the values of the phases ϕ_k and ϕ_j. Thus, we construct orthogonal frequency-division multiplexed (OFDM) signals. In other words, OFDM is a special type of multicarrier modulation in which the subcarriers of the corresponding subchannels are mutually orthogonal, as defined in Equation 11.2–10.

Multicarrier modulation (OFDM) is widely used in both wireline and radio channels. For example, OFDM has been adopted as a standard for digital audio broadcast applications and wireless local area networks based on the IEEE 802.11 standard.

A particular suitable application of OFDM is in digital transmission over copper wire subscriber loops. The typical channel attenuation characteristics for such subscriber lines are illustrated in Figure 11.2–3. We observe that the attenuation increases rapidly as a function of frequency. This characteristic makes it extremely difficult to

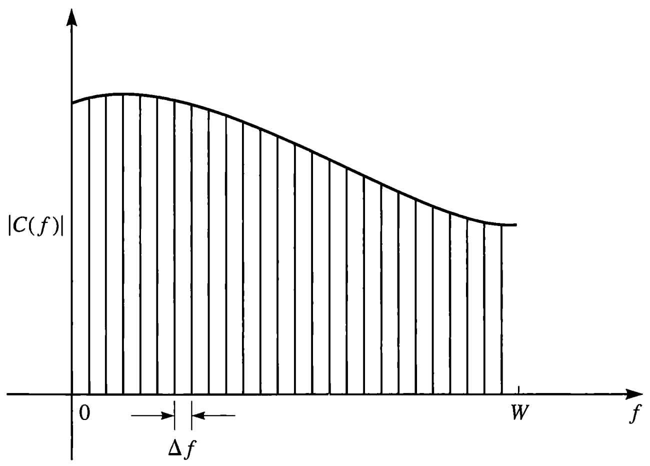


FIGURE 11.2–2
Subdivision of the channel bandwidth W into narrowband subchannels of equal width Δf.

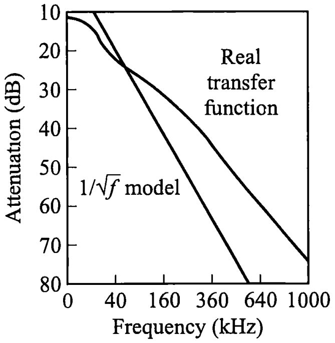


FIGURE 11.2–3
Attenuation characteristic of a 24-gauge 12,000-ft polyethylene-insulated cable loop. [*From Werner (1991)* © *IEEE.*]

achieve a high transmission rate with a single modulated carrier and an equalizer at the receiver. The ISI penalty in performance is very large. On the other hand, OFDM with optimum power distribution provides the potential for a higher transmission rate.

The dominant noise in transmission over subscriber lines is crosstalk interference from signals carried on other telephone lines located in the same cable. The power distribution of this type of noise is also frequency-dependent, which can be taken into consideration in the allocation of the available transmitted power.

A design procedure for a multicarrier QAM system for a nonideal linear filter channel has been given by Kalet (1989). In this procedure, the overall bit rate is maximized, through the design of an optimal power division among the subcarriers and an optimum selection of the number of bits per symbol (sizes of the QAM signal constellations) for each subcarrier, under an average power constraint and under the constraint that the symbol error probabilities for all subcarriers are equal.

11.2–4 Modulation and Demodulation in an OFDM System

In an OFDM system with N subchannels, the symbol rate $1/T$ is reduced by a factor of N relative to the symbol rate on a single carrier system that employs the entire bandwidth W and transmits data at the same rate as OFDM. Hence, the symbol interval in the OFDM system is $T = NT_s$, where T_s is the symbol interval in the single-carrier system. By selecting N to be sufficiently large, the symbol interval T can be made significantly larger than the time duration of the channel-time dispersion. Thus, intersymbol interference can be made arbitrarily small through the selection of N. In other words, each subchannel appears to have a fixed frequency response $C(f_k)$, $k = 0, 1, \ldots, N-1$.

Suppose that each subcarrier is modulated with M-ary QAM. Then the signal on the kth subcarrier may be expressed as

$$\begin{aligned} u_k(t) &= \sqrt{\frac{2}{T}} A_{ki} \cos 2\pi f_k t - \sqrt{\frac{2}{T}} A_{kq} \sin 2\pi f_k t \\ &= \operatorname{Re}\left[\sqrt{\frac{2}{T}} A_k e^{j\theta_k}\, e^{j2\pi f_k t}\right] \qquad (11.2\text{–}11) \\ &= \operatorname{Re}\left[\sqrt{\frac{2}{T}} X_k\, e^{j2\pi f_k t}\right] \end{aligned}$$

where $X_k = A_k e^{j\theta_k}$ is the signal point from the QAM signal constellation that is transmitted on the kth subcarrier, $A_k = \sqrt{A_{ki}^2 + A_{kq}^2}$, and $\theta_k = \tan^{-1}(A_{kq}/A_{ki})$. The energy per symbol $\mathcal{E}_s$ has been absorbed into $\{X_k\}$.

When the number of subchannels is large, so that the subchannels are sufficiently narrowband, each subchannel can be characterized by a fixed frequency response $C(f_k)$, $k = 0, 1, \ldots, N-1$. In general, $C(f_k)$ is complex-valued and may be expressed as

$$C(f_k) = C_k = |C_k| e^{j\phi_k} \tag{11.2–12}$$

Hence, the received signal on the kth subchannel is

$$\begin{aligned} r_k(t) &= \sqrt{\frac{2}{T}} |C_k| A_{kc} \cos(2\pi f_k t + \phi_k) + \sqrt{\frac{2}{T}} |C_k| A_{ks} \sin(2\pi f_k t + \phi_k) + n_k(t) \\ &= \operatorname{Re}\left[\sqrt{\frac{2}{T}} C_k X_k \, e^{j2\pi f_k t}\right] + n_k(t) \end{aligned} \tag{11.2–13}$$

where $n_k(t)$ represents the additive noise in the kth subchannel. We assume that $n_k(t)$ is zero-mean Gaussian and spectrally flat across the bandwidth of the kth subchannel. We also assume that the channel parameters $|C_k|$ and ϕ_k are known at the receiver. (These parameters are usually estimated by initially transmitting the unmodulated carrier cos $2\pi f_k t$ and observing the received signal $|C_k| \cos(2\pi f_k t + \phi_k)$.)

The demodulation of the received signal in the kth subchannel may be accomplished by cross-correlating $r_k(t)$ with the two basis functions, based on knowledge of the carrier phase $\{\phi_k\}$ at the receiver,

$$\begin{aligned} \psi_1(t) &= \sqrt{\frac{2}{T}} \cos(2\pi f_k t + \phi_k), \qquad 0 \le t \le T \\ \psi_2(t) &= -\sqrt{\frac{2}{T}} \sin(2\pi f_k t + \phi_k), \qquad 0 \le t \le T \end{aligned} \tag{11.2–14}$$

and sampling the output of the cross-correlators at $t = T$. Thus, we obtain the received signal vector

$$y_k = (|C_k| A_{ki} + \eta_{kr}, \quad |C_k| A_{kq} + \eta_{ki}) \tag{11.2–15}$$

which can also be expressed as the complex number

$$Y_k = |C_k| X_k + \eta_k \tag{11.2–16}$$

where $\eta_k = \eta_{kr} + j\eta_{ki}$ represents the additive noise.

The scaling of the transmitted symbol by the channel gain $|C_k|$ can be removed by dividing Y_k by $|C_k|$. Thus, we obtain

$$Y_k' = Y_k / |C_k| = X_k + \eta_k' \tag{11.2–17}$$

where $\eta_k^{'} = \eta_k/|C_k|$. The normalized variable $Y_k^{'}$ is passed to the detector, which computes the distance metrics between $Y_k^{'}$ and each of the possible signal points in the QAM signal constellation and selects the signal point resulting in the smallest distance.

From this description, it is clear that two cross-correlators or two matched filters are required to demodulate the received signal in each subchannel. Therefore, if the OFDM signal consists of N subchannels, the implementation of the OFDM demodulator requires a parallel bank of $2N$ cross-correlators or $2N$ matched filters. Furthermore, the modulation process for generating the OFDM signal can also be viewed as exciting a bank of $2N$ parallel filters with symbols taken from an M-ary QAM signal constellation.

The bank of $2N$ parallel filters that generates the modulated signal at the transmitter and demodulates the received signal is equivalent to the computation of the discrete Fourier transform (DFT) and its inverse. Since an efficient computation of the DFT is the fast Fourier transform (FFT) algorithm, a more efficient implementation of the modulation and demodulation processes when N is large, e.g., $N > 32$, is by means of the FFT algorithm. In the next section, we describe the implementation of the modulator and demodulator in an OFDM system that uses the FFT algorithm to compute the DFT.

Since the signals transmitted on the N subchannels of the OFDM system are synchronized, the received signals on any pair of subchannels are orthogonal over the interval $0 \leq t \leq T$. If the subchannel gains $|C_k|, 0 \leq k \leq N-1$, are sufficiently different across the channel bandwidth, subchannels that yield a higher SNR due to a lower attenuation can be modulated to carry more bits per symbol than subcarriers that yield a lower SNR (high attenuation). Consequently, QAM with different constellation sizes can be used on the different subchannels of an OFDM system. This assignment of different constellation sizes to different subchannels is generally done in practice.

11.2–5 An FFT Algorithm Implementation of an OFDM System

In this section we describe a multicarrier communication system that employs the fast Fourier transform algorithm to synthesize the signal at the transmitter and to demodulate the received signal at the receiver. The FFT is simply the efficient computational tool for implementing the DFT.

Figure 11.2–4 illustrates a block diagram of a multicarrier communication system. A serial-to-parallel buffer segments the information sequence into frames of N_f bits. The N_f bits in each frame are parsed into $\tilde{N}$ groups, where the ith group is assigned b_i bits, and

$$\sum_{i=1}^{\tilde{N}} b_i = N_f \tag{11.2–18}$$

Each group may be encoded separately, so that the number of output bits from the encoder for the ith group is $n_i \geq b_i$.

It is convenient to view the multicarrier modulation as consisting of $\tilde{N}$ independent QAM channels, each operating at the same symbol rate $1/T$, but each channel having a distinct QAM constellation; i.e., the ith channel will employ $M = 2^{b_i}$ signal points.

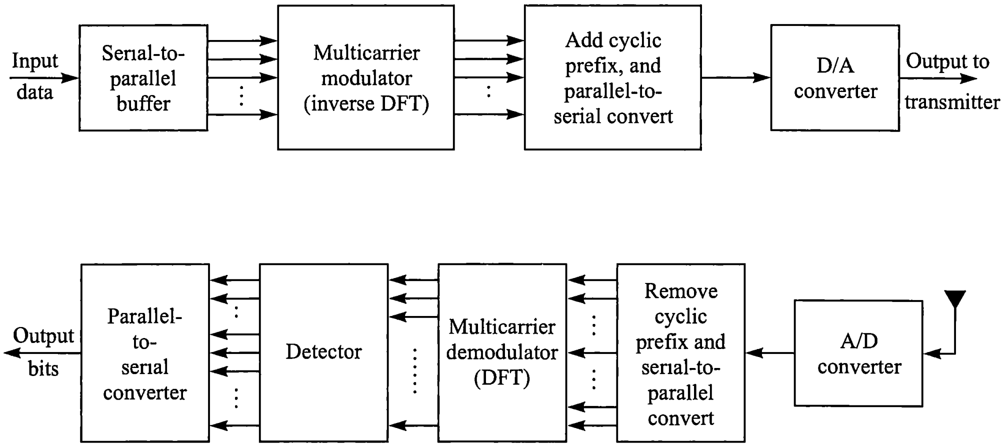

Input data
Serial-to-parallel buffer
Multicarrier modulator (inverse DFT)
Add cyclic prefix, and parallel-to-serial convert
D/A converter
Output to transmitter
Output bits
Parallel-to-serial converter
Detector
Multicarrier demodulator (DFT)
Remove cyclic prefix and serial-to-parallel convert
A/D converter

FIGURE 11.2–4
Multicarrier communication system.

We denote the complex-valued signal points corresponding to the information symbols on the subchannels by $X_k,\ k = 0, 1, \ldots, \tilde{N} - 1$. To modulate the $\tilde{N}$ subcarriers by the information symbols $\{X_k\}$, we employ the inverse DFT (IDFT).

However, if we compute the $\tilde{N}$-point IDFT of $\{X_k\}$, we obtain a complex-valued time series, which is not equivalent to $\tilde{N}$ QAM-modulated subcarriers. Instead, we create $N = 2\tilde{N}$ information symbols by defining

$$X_{N-k} = X_k^*, \qquad k = 1, \ldots, \tilde{N} - 1 \qquad (11.2\text{–}19)$$

and $X_0 = \text{Re}(X_0)$, $X_{\tilde{N}} = \text{Im}(X_0)$. Thus, the symbol X_0 is split into two parts, both real. Then the N-point IDFT yields the real-valued sequence

$$x_n = \frac{1}{\sqrt{N}} \sum_{k-0}^{N-1} X_k\, e^{j2\pi nk/N}, \qquad n = 0, 1, \ldots, N - 1 \qquad (11.2\text{–}20)$$

where $1/\sqrt{N}$ is simply a scale factor.

The sequence $\{x_n, 0 \le n \le N - 1\}$ corresponds to the samples of the sum $x(t)$ of $\tilde{N}$ subcarrier signals, which is expressed as

$$x(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k\, e^{j2\pi kt/T}, \qquad 0 \le t \le T \qquad (11.2\text{–}21)$$

where T is the symbol duration. We observe that the subcarrier frequencies are $f_k = k/T,\ k = 0, 1, \ldots, \tilde{N}$. Furthermore, the discrete-time sequence $\{x_n\}$ in Equation 11.2–20 represents the samples of $x(t)$ taken at times $t = nT/N$ where $n = 0, 1, \ldots, N - 1$.

The computation of the IDFT of the data $\{X_k\}$ as given in Equation 11.2–20 may be viewed as multiplication of each data point X_k by a corresponding vector

$$\boldsymbol{v}_k = [v_{k0} \quad v_{k1} \ \ldots \ v_{k(N-1)}] \qquad (11.2\text{–}22)$$

where

$$v_{kn} = \frac{1}{\sqrt{N}} e^{j(2\pi/N)kn} \qquad (11.2\text{–}23)$$

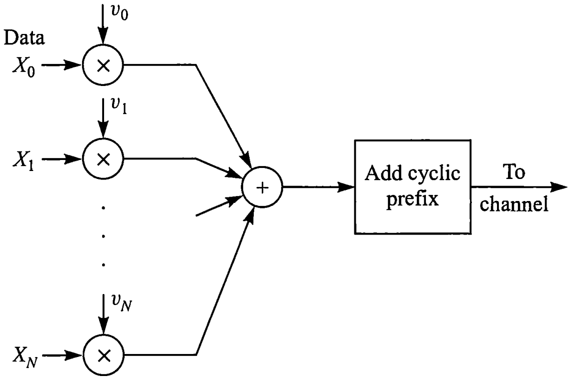


FIGURE 11.2–5
Signal synthesis for multicarrier modulation based on inverse DFT.

as illustrated in Figure 11.2–5. In any case, the computation of the DFT is performed efficiently by the use of the FFT algorithm.

In practice, the signal samples $\{x_n\}$ are passed through a digital-to-analog (D/A) converter whose output, ideally, would be the signal waveform $x(t)$. The output of the channel is the waveform

$$r(t) = x(t) * c(t) + n(t) \tag{11.2–24}$$

where $c(t)$ is the impulse response of the channel and $*$ denotes convolution. By selecting the bandwidth Δf of each subchannel to be very small, the symbol duration $T = 1/\Delta f$ is large compared with the channel time dispersion. To be specific, let us assume that the channel dispersion spans $\nu + 1$ signal samples where $\nu \ll N$. One way to avoid the effect of ISI is to insert a time guard band of duration $\nu T/N$ between transmissions of successive blocks.

An alternative method that avoids ISI is to append a cyclic prefix to each block of N signal samples $\{x_0, x_1, \ldots, x_{N-1}\}$. The cyclic prefix for this block of samples consists of the samples $x_{N-\nu}, x_{N-\nu+1}, \ldots, x_{N-1}$. These new samples are appended to the beginning of each block. Note that the addition of the cyclic prefix to the block of data increases the length of the block to $N + \nu$ samples, which may be indexed from $n = -\nu, \ldots, N - 1$, where the first ν samples constitute the prefix. Then if $\{c_n, 0 \le n \le \nu\}$ denotes the sampled channel impulse response, its convolution with $\{x_n, -\nu \le n \le N - 1\}$ produces $\{r_n\}$, the received sequence. We are interested in the samples of $\{r_n\}$ for $0 \le n \le N-1$, from which we recover the transmitted sequence by using the N-point DFT for demodulation. Thus, the first ν samples of $\{r_n\}$ are discarded.

From a frequency-domain viewpoint, when the channel impulse response is $\{c_n, 0 \le n \le \nu\}$, its frequency response at the subcarrier frequencies $f_k = k/N$ is

$$C_k = C\left(\frac{2\pi k}{N}\right) = \sum_{n=0}^{\nu} c_n\, e^{-j2\pi nk/N} \tag{11.2–25}$$

Because the cyclic prefix serves as a time guard band against interference, successive blocks (frames) of the transmitted information sequence do not interfere and, hence, the demodulated sequence may be expressed as

$$\tilde{X}_k = C_k X_k + \eta_k, \qquad k = 0, 1, \ldots, N - 1 \tag{11.2–26}$$

where $\{\tilde{X}_k\}$ is the output of the N-point DFT demodulator and η_k is the additive noise corrupting the signal. We note that by selecting $N \gg \nu$, the rate loss due to the cyclic prefix can be rendered negligible.

As shown in Figure 11.2–4, the information is demodulated by computing the DFT of the received signal after it has been passed through an analog-to-digital (A/D) converter. The DFT computation may be viewed as a multiplication of the received signal samples $\{r_n\}$ from the A/D converter by $\boldsymbol{v}_k^*$, where $\boldsymbol{v}_k$ is defined in Equation 11.2–22. As in the case of the modulator, the DFT computation at the demodulator is performed efficiently by use of the FFT algorithm.

It is simple matter to estimate and compensate for the channel factors $\{C_k\}$ prior to passing the data to the detector and decoder. A training signal consisting of either a known modulated sequence on each of the subcarriers or unmodulated subcarriers may be used to measure the $\{C_k\}$ at the receiver. If the channel parameters vary slowly with time, it is also possible to track the time variations by using the decisions at the output of the detector or the decoder, in a decision-directed fashion. Thus, the multicarrier system can be rendered adaptive.

By measuring the SNR in each subchannel, one can optimize the transmission rate by allocating the average transmitted power and the number of bits to be carried by each subcarrier. The SNR per subchannel is defined as

$$\mathrm{SNR}_k = \frac{TP_k|C_k|^2}{\sigma_{nk}^2} \tag{11.2–27}$$

where T is the symbol duration, P_k is the average power allocated to the kth subchannel, $|C_k|^2$ is the magnitude squared of the frequency response of the kth subchannel, and σ_{nk}^2 is the variance of the noise in the kth subchannel. Based on these SNR measurements, the capacity of each subchannel may be determined as described in Section 11.2–2. Furthermore, system performance may be optimized by selecting the bit and power allocation for each subchannel as described below and in the papers by Chow et al. (1995) and Fischer and Huber (1996).

Multicarrier QAM of the type described above has been implemented for a variety of applications, including high-speed transmission over telephone lines, such as digital subscriber lines.

Other types of implementation besides the DFT are possible. For example, a digital filter bank that basically performs the DFT may be substituted for the FFT-based implementation when the number of subcarriers is small, e.g., $N \leq 32$. For a large number of subcarriers, e.g., $N > 32$, the FFT-based systems are computationally more efficient.

11.2–6 Spectral Characteristics of Multicarrier Signals

Although the signals transmitted on the subcarriers of an OFDM system are mutually orthogonal in the time domain, these signals have significant overlap in the frequency

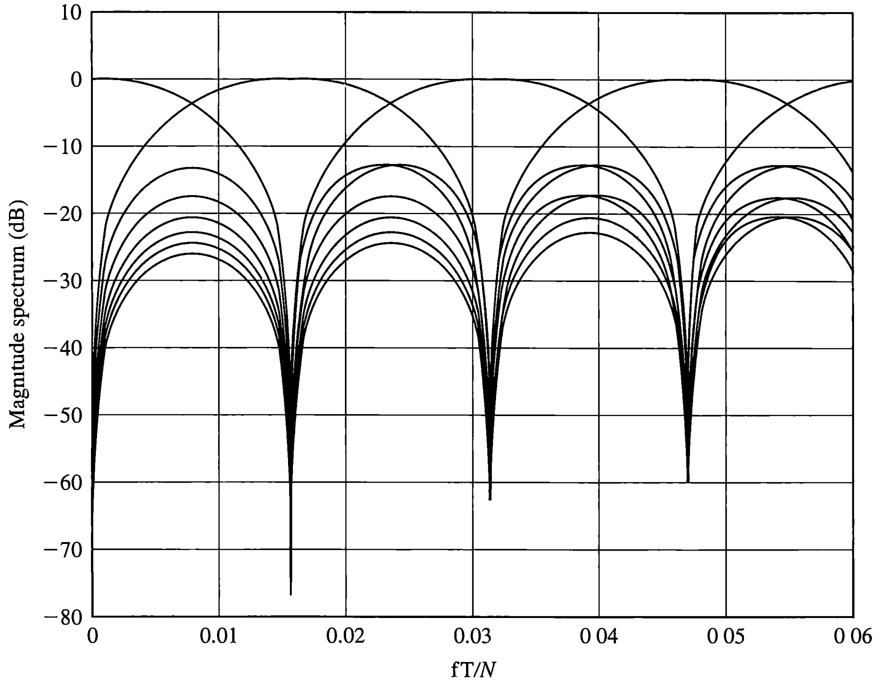


FIGURE 11.2–6
An example of the magnitude of the frequency response of adjacent subchannel filters in OFDM system for $f \in (0, 0.06\frac{N}{T})$ and $N = 64$. [*From Cherubini et al. (2002) IEEE.*]

domain. This can be observed by computing the Fourier transform of the signal

$$\begin{aligned} u_k(t) &= \text{Re}\left[\sqrt{\frac{2}{T}} X_k\, e^{j2\pi f_k t}\right] \\ &= \sqrt{\frac{2}{T}} A_k \cos(2\pi f_k t + \theta_k), \qquad 0 \le t \le T \end{aligned} \tag{11.2–28}$$

for several values of k. Figure 11.2–6 illustrates the magnitude spectrum $|U_k(f)|$ for several adjacent subcarriers. Note the large spectral overlap of the main lobes. Also note that the first sidelobe in the spectrum is only 13 dB down from the main lobe. Hence, there is a significant amount of spectral overlap among the signals transmitted on different subcarriers. Nevertheless, these signals are orthogonal when transmitted synchronously in time.

The large spectral overlap of the OFDM signals has various ramifications when the communication channel is a radio channel and the receiving terminal is mobile, as in the case of cellular radio communications. In such mobile radio communications, the transmitted signal is imparted with Doppler frequency shifts or Doppler spreading, which destroys the orthogonality among the subcarriers and, as a consequence, results in interchannel interference (ICI). The ICI produces a significant degradation in the performance (error probability) of the OFDM system. The degree of performance degradation is proportional to the speed at which the receiving terminal is moving. In general, the degradation is small when the terminal is moving at pedestrian speed.

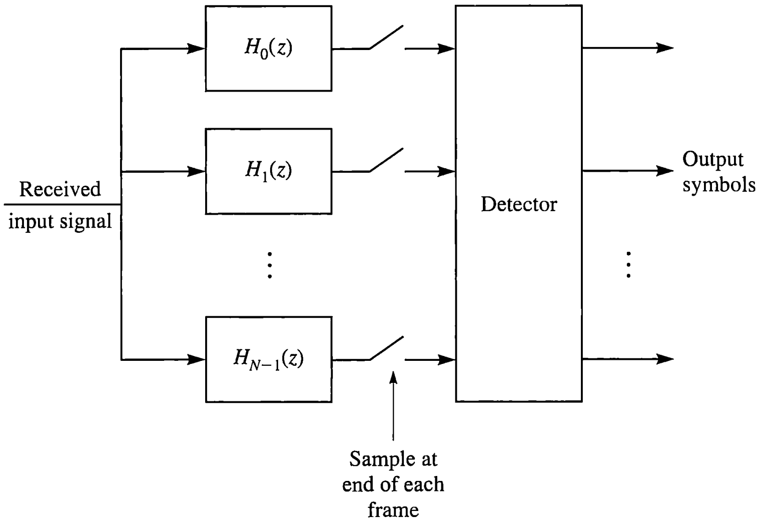


FIGURE 11.2–7
Filter bank implementation of OFDM receiver.

This is the case, for example, in wireless LANs that employ OFDM signals with large $(M = 64)$ QAM signal constellations.

The detrimental effects of ICI in a multicarrier system, such as OFDM, can be significantly reduced by employing a bank of parallel filters in the implementation of the system, as illustrated in Figure 11.2–7. In such an implementation, the prototype filter $H_0(f)$ and, hence, its frequency-shifted versions $H_k(f) = H_0(f - k/T)$ are designed to have sharp cutoff frequency-response characteristics. Consequently, a Doppler frequency spread that is small compared to $1/2T$, or equivalently, compared to the bandwidth of the prototype filter $H_0(f)$, will result in negligible ICI. For example, Figure 11.2–8 illustrates the frequency-response characteristics in such a filter bank implementation. Note that the filter sidelobes are approximately 70 dB below the main lobe, and the spectral overlap between adjacent filters is negligible. Such filter characteristics provide significant immunity against ICI that may be encountered in mobile radio communication environments.

The price paid for achieving this immunity to ICI caused by Doppler spreading is the added complexity in the implementation of the filters $\{H_k(f)\}$ at the transmitter and the receiver. An efficient implementation for the filter bank, based on multirate digital signal processing methods, has been described in the papers by Cherubini et al. (2000, 2002). The resulting filter bank implementation of the multicarrier system is called *filtered multitone* (FMT) *modulation*. The spectral characteristics shown in Figure 11.2–8 correspond to filter frequency responses in an FMT multicarrier modulation system.

11.2–7 Bit and Power Allocation in Multicarrier Modulation

We now consider a bit and power allocation procedure to optimize the performance of a multicarrier system transmitting over a linear time-invariant channel with AWGN. We assume that there are $\tilde{N}$ subcarriers and that the modulation on each subcarrier is

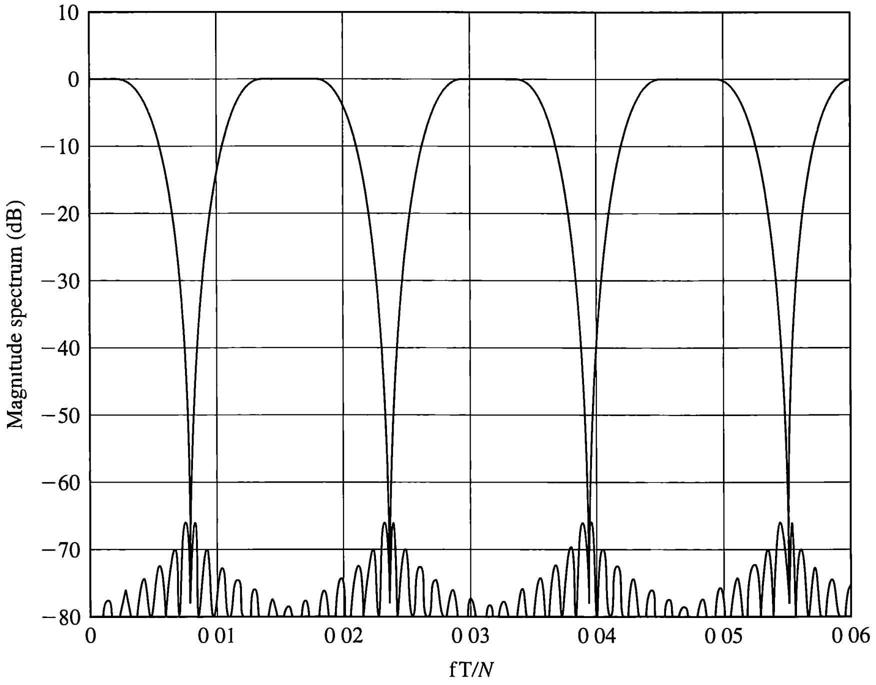


FIGURE 11.2–8
An example of the magnitude of the frequency response of adjacent subchannel filters in an FMT system for $f \in (0, 0.06\frac{N}{T})$ and design parameters $N = 64$. [*From Cherubini et al. (2002) IEEE.*]

QAM, where $M_i = 2^{b_i}$ is the constellation size and b_i is the number of bits transmitted on the ith subcarrier in the frame interval of T seconds. Thus, the total bit rate is

$$R_b = \frac{1}{T}\sum_{i=1}^{\tilde{N}} b_i \tag{11.2–29}$$

The power allocated to the ith subcarrier is P_i, and the total transmitted power is

$$P = \sum_{i=1}^{\tilde{N}} P_i \tag{11.2–30}$$

which is constrained to be a fixed value.

The bandwidth of each subchannel is assumed to be sufficiently narrow that the complex-valued channel gain $C(f_i)$ is constant across the frequency band of the ith subchannel. For convenience, we also assume that the spectral density of the additive Gaussian noise in the $\tilde{N}$ subchannels is identical.

In selecting the bit and power allocation among the $\tilde{N}$ subchannels, our objective is to maximize the bit rate R_b for a specified error probability that is the same across the $\tilde{N}$ subchannels. It is convenient to use the symbol error probability for QAM as the performance index and to focus on the low-error-rate (high-SNR) region. The symbol error probability for QAM at low error rates is well approximated by the expression

$$P_e \approx 4Q\left(\sqrt{\frac{3P_i|C_i|^2}{N_0(M_i-1)}}\right) \tag{11.2–31}$$

where P_e is the desired symbol error probability and $C_i \equiv C(f_i)$. The multiplier in front of the Q function represents the number of nearest neighbors in a rectangular QAM signal constellation. Therefore, P_i and M_i are selected such that

$$Q\left(\sqrt{\frac{3P_i|C_i|^2}{N_0(M_i-1)}}\right) = \frac{P_e}{4} \tag{11.2–32}$$

It has been shown by Kalet (1989) that transmitting equal power across all subchannels for which $|C_i|^2/N_0$ is sufficiently large to support at least an $M = 4$ signal constellation at the desired low symbol error probability results in near optimum performance. Hence, we may begin by allocating equal power among the subchannels and deleting all subcarriers which cannot support at least an $M = 4$ signal constellation at the desired error probability. Then we allocate the total transmit power equally among the remaining subchannels and compute the value of M_i that satisfies the desired error probability given by Equation (11.2–32).

At this point, we may simply truncate the values of $\{M_i\}$ to $\{\tilde{M}_i\}$ such that

$$b_i = \log_2 \tilde{M}_i, \qquad i = 1, 2, \ldots, N \tag{11.2–33}$$

are integers. However, when the number of subchannels is large, this simple allocation procedure may result in a significant loss in rate. Alternatively, we may use the unquantized value of each M_i that satisfies the desired symbol error probability and either round up to the next-higher power of 2 or truncate to the next-lower power of 2, if the fractional part of the bit $b_i = \log_2 M_i$ is greater than 1/2 or lower than 1/2, respectively. The allocated power for each subchannel is then adjusted accordingly to satisfy the desired error probability. This power allocation procedure may be performed sequentially, beginning with the subchannel having the largest $|C_i|^2/N_0$, where at each step the remaining power is allocated equally among the remaining subchannels. Thus, the total power allocation is kept constant.

As an example, let us consider high-speed digital transmission over wirelines that connect a telephone subscriber's premises to a telephone central office. These wireline channels typically consist of unshielded twisted-pair wire and are commonly called the *subscriber local loop*. The desire to provide high-speed Internet access to homes and businesses over the telephone subscriber loop has resulted in the development of a standard for digital transmission based on OFDM with QAM as the basic modulation method on each of the subcarriers.

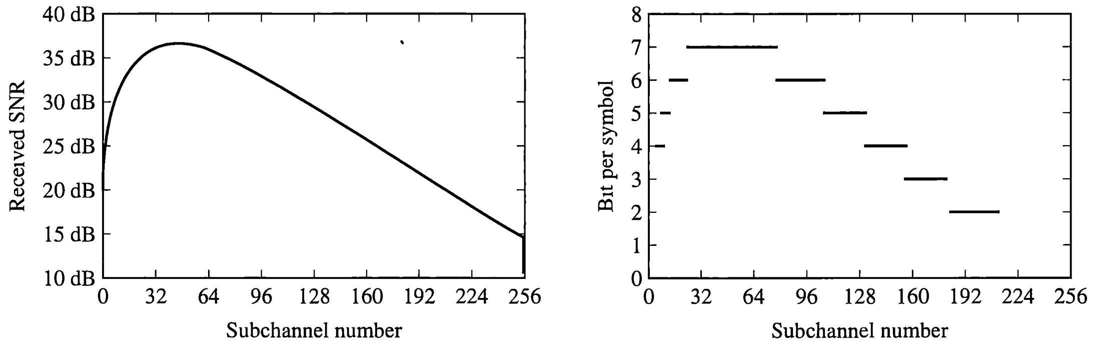
The usable bandwidth of a twisted-pair subscriber loop wire is primarily limited by the distance between the subscriber and the central telephone office, i.e., the length of the wire, and by crosstalk interference from other lines in the same cable. For example, a 3-km twisted-pair wireline may have a usable bandwidth of approximately 1.2 MHz. Since the need for high-speed digital transmission is usually in the direction from the central office to the subscriber (the downlink) and the bandwidth is relatively small, the major part of the bandwidth is allocated to the downlink. Consequently, the digital transmission on the subscriber loop is asymmetric, and this transmission mode is called *ADSL* (*asymmetric digital subscriber line*).

In the ADSL standard, the downlink and the uplink maximum data rates are specified as 6.8 Mbps and 640 kbps, respectively, for subscriber lines of approximately 12,000 ft in length, and 1.544 Mbps and 176 kbps, respectively, for subscriber lines of approximately 18,000 ft in length. The low part of the frequency band (0–25 kHz) is reserved for the telephone voice transmission, which requires a nominal bandwidth of 4 kHz. Hence, the frequency band of the subscriber line is separated into two frequency bands via two filters (lowpass and highpass) that have cutoff frequencies of 25 kHz. Thus, the low-end frequency for digital transmission is 25 kHz. The ADSL standard specifies that the frequency range of 25 kHz to 1.1 MHz must be subdivided into 256 parallel OFDM subchannels Hence, the size of the DFT and IDFT in the system implementation shown in Figure 11.2–4 is $N = 512$. A sampling rate $f_s = 2.208$ MHz is specified, so that the high-end frequency in the signal spectrum is $f_s/2 = 1.104$ MHz. The frequency spacing between two adjacent subcarriers is $\Delta f = 1.104 \times 10^6/256 = 4.3125$ kHz. The channel time dispersion is suppressed by using a cyclic prefix of $N/16 = 32$ samples.

By measuring the signal-to-noise ratio (SNR) for each subchannel at the receiver and communicating this information to the transmitter via the uplink, the transmitter can select the QAM constellation size in bits per symbol to achieve a desired error probability in each subchannel. The ADSL standard specifies a minimum bit load of 2 bits per subchannel, which corresponds to QPSK modulation. If a subchannel cannot support QPSK at the desired error probability, no information is transmitted over that subchannel. As an example, Figure 11.2–9 illustrates the received SNR as measured by the receiver for each subchannel and the corresponding number of bits per symbol selected from a QAM signal constellation. Note that the SNR in subchannels 220–256 is too low to support QPSK modulation; hence, no data are transmitted on these subchannels. ADSL channel characteristics and the design of OFDM modems based on the ADSL standard are treated in detail in the books by Bingham (2000) and Starr et al. (1999). The use of OFDM with variable size QAM signal constellations for each of the subcarriers is sometimes called *discrete multitone* (DMT) *modulation.*

11.2–8 Peak-to-Average Ratio in Multicarrier Modulation

A major problem with multicarrier modulation is the relatively high peak-to-average ratio (PAR) that is inherent in the transmitted signal. In general, large signal peaks occur in the transmitted signal when the signals in many of the various subchannels



(a) (b)

FIGURE 11.2–9
Example of a DSL frequency response and bit allocation on the OFDM subchannels.

add constructively in phase. Such large signal peaks may result in clipping of the signal voltage in a D/A converter when the multicarrier signal is synthesized digitally, and/or it may saturate the power amplifier and thus cause intermodulation distortion in the transmitted signal. When the number N of subcarriers is large, the central limit theorem may be used to model the combined signal on the N subchannels as a zero-mean Gaussian random process. In such a model, the voltage PAR is proportional to $\sqrt{N}$.

To avoid intermodulation distortion, it is common to reduce the power in the transmitted signal and thus operate the power amplifier at the transmitter in the linear operating range. This power reduction or "power backoff" results in inefficient operation of the communication system. For example, if the PAR is 10 dB, the power backoff may be as much as 10 dB to avoid intermodulation distortion.

Various methods have been devised to reduce the PAR in multicarrier systems. One of the simplest methods is to insert different phase shifts in each of the subcarriers. These phase shifts can be selected pseudorandomly, or by means of some algorithm, to reduce the PAR. For example, we may have a small set of N stored pseudorandomly selected phase shifts which can be used when the PAR in the modulated subcarriers is large. The information on which set of pseudorandom phase shifts is used in any signal interval can be transmitted to the receiver on one of the N subcarriers. Alternatively, a single set of pseudorandom phase shifts may be employed, where this set is found via computer simulation to reduce the PAR to an acceptable level over the ensemble of possible transmitted data symbols on the N subcarriers.

Another method that can be used to reduce the PAR is to modulate a small subset of the subcarriers with dummy symbols which are selected to reduce the PAR. Since the dummy symbols do not have to be constrained to take amplitude and phase values from a specified signal constellation, the design of the dummy symbols is very flexible. The subcarriers carrying dummy symbols may be distributed across the frequency band. Since modulating subcarriers with dummy symbols results in a lower throughput in data rate, it is desirable to employ only a small percentage of the total subcarriers for this purpose.

As an alternative to allocating subcarriers that are modulated with dummy symbols, one may select a subset of subcarriers that already carry data and expand the signal constellation in such a manner that the data can be correctly detected at the receiver by use of a modulo-q operation, where q is an appropriate integer. For example, if rectangular 16-point QAM is used as the modulation of each subcarrier, a minimally expanded signal constellation for a subset of subcarriers may consist of a 32-point signal constellation that includes the 16 additional points adjacent to the outer points in the original constellation. When the PAR of the original signal constellation exceeds a predetermined amount, the signal point on a selected subcarrier is replaced by a signal point from the minimally expanded set such that the PAR is reduced. This approach may require several iterations using a different subcarrier each time to reduce the PAR to a desired value. The interested reader may refer to the paper by Tellado and Cioffi (1998), which treats this method.

In a digitally synthesized multicarrier signal, the PAR may be kept within a specified limit by clipping the signal at the D/A converter. The clipping generally distorts the signal at the transmitter and hence degrades the performance at the receiver. The effect of clipping on the probability of error at the detector in an OFDM system has been evaluated by Bahai and Saltzberg (1999). If the clipping occurs infrequently, the occasional errors may be corrected by introducing a suitable error-correcting code.

Because of its practical importance, the problem of PAR reduction in multicarrier systems has been investigated by many people, and methods other than the ones described above have been considered. The interested reader may refer to the papers by Boyd (1986), Popovic (1991), Jones et al. (1994), Wilkinson and Jones (1995), Wulich (1996), Li and Cimini (1997), Friese (1997), Müller et al. (1997), Tellado and Cioffi (1998), Wulich and Goldfeld (1999), Tarokh and Jafarkhani (2000), Peterson and Tarokh (2000), and Wunder and Boche (2003).

11.2–9 Channel Coding Considerations in Multicarrier Modulation

In single-carrier systems, channel coding is performed in the time domain. That is, the coded bits or symbols span multiple signal or symbol intervals. In multicarrier communication systems, such as OFDM, the frequency domain provides an additional dimension in which channel coding can be applied to achieve immunity against noise and other interference.

One possible channel coding approach is to encode the information bits on each subcarrier separately (time-domain channel coding) using either a block code, or a convolutional code, or by employing trellis-coded modulation (TCM). In such a time-domain coding approach, the coded bits or symbols span multiple OFDM (multicarrier) frames. There are basically two disadvantages with time-domain channel coding for multicarrier communication systems. One is the encoding/decoding complexity involved in the operation of N parallel encoders/decoders for the N subchannels. The second is the latency (decoding delay) inherent in the decoding of the data on the N subcarriers over multiple frames. For example, the decoding delay for a code that spans K frames is KN_f bits, where N_f is the number of information bits per frame.

The decoding delay can be minimized by designing the channel code to span the bits across the subchannels for a single OFDM (multicarrier) frame. In such a frequency-domain coding approach we may employ a block code, or a convolutional code, or TCM. If additional delay beyond a single frame is tolerable, the channel code may be designed to span multiple OFDM frames. The advantage of this approach to channel coding for multicarrier communication systems is that a single encoder and decoder can be employed in the system, thus simplifying the system implementation.

Although the channel coding methods for multicarrier modulation described above focused on simple coding techniques (block coding, convolutional coding, TCM), they are easily extended to concatenated coding and turbo coding methods.

11.3

BIBLIOGRAPHICAL NOTES AND REFERENCES

Multichannel signal transmission is commonly used on time-varying channels to overcome the effects of signal fading. This topic is treated in some detail in Chapter 13, where we provide a number of references to published work. Of particular relevance to the treatment of multichannel digital communications given in this chapter are the two publications by Price (1962a, b).

There is a large amount of literature on multicarrier digital communication systems. Such systems have been implemented and used for over 35 years. One of the earliest systems, described by Doeltz et al. (1957) and called Kineplex, was used for digital transmission in the HF band. Other early work on multicarrier system design has been reported in the papers by Chang (1966) and Saltzberg (1967). The use of the DFT for modulation and demodulation of multicarrier systems was proposed by Weinstein and Ebert (1971).

Of particular interest in recent years is the use of multicarrier digital transmission for data, facsimile, and video on a variety of channels, including the narrowband (4 kHz) switched telephone network, the 48-kHz group telephone band, digital subscriber lines, cellular radio, and audio broadcast. The interested reader may refer to the many papers in the literature. We cite as examples the papers by Hirosaki (1981), Hirosaki et al. (1986), Chow et al. (1991), and the survey paper by Bingham (1990). The paper by Kalet (1989) gives a design procedure for optimizing the rate in a multicarrier QAM system given constraints on transmitter power and channel characteristics. Finally, we cite the book by Vaidyanathan (1993) and the papers by Tzannes et al. (1994) and Rizos et al. (1994) for a treatment of multirate digital filter banks, and the books by Starr et al. (1999) and Bingham (2000) on the application of multicarrier modulation for digital transmission on digital subscriber lines.

PROBLEMS

11.1 $X_1, X_2, \ldots, X_N$ are a set of N statistically independent and identically distributed real Gaussian random variables with moments $E(X_i) = m$ and $\text{var}(X_i) = \sigma^2$.

a. Define

$$U = \sum_{n=1}^{N} X_n$$

Evaluate the SNR of U, which is defined as

$$(\text{SNR})_U = \frac{[E(U)]^2}{2\sigma_U^2}$$

where σ_U^2 is the variance of U.

b. Define

$$V = \sum_{n=1}^{N} X_n^2$$

Evaluate the SNR of V, which is defined as

$$(\text{SNR})_V = \frac{[E(V)]^2}{2\sigma_V^2}$$

where σ_V^2 is the variance of V.

c. Plot $(\mathrm{SNR})_U$ and $(\mathrm{SNR})_V$ versus m^2/σ^2 on the same graph and, thus, compare the SNRs graphically.

d. What does the result in (c) imply regarding coherent detection and combining versus square-law detection and combining of multichannel signals?

11.2 A binary communication system transmits the same information on two diversity channels. The two received signals are

$$r_1 = \pm\sqrt{\mathcal{E}_b} + n_1$$
$$r_2 = \pm\sqrt{\mathcal{E}_b} + n_2$$

where $E(n_1) = E(n_2) = 0$, $E\left(n_1^2\right) = \sigma_1^2$ and $E\left(n_2^2\right) = \sigma_2^2$, and n_1 and n_2 are uncorrelated Gaussian variables. The detector bases its decision on the linear combination of r_1 and r_2, i.e.,

$$r = r_1 + kr_2$$

a. Determine the value of k that minimizes the probability of error.

b. Plot the probability of error for $\sigma_1^2 = 1$, $\sigma_2^2 = 3$, and either $k = 1$ or k is the optimum value found in (a). Compare the results.

11.3 Assess the cost of the cyclic prefix (used in multicarrier modulation to avoid ISI) in terms of

a. Extra channel bandwidth.

b. Extra signal energy.

11.4 Let $x(n)$ be a finite-duration signal with length N and let $X(k)$ be its N-point DFT. Suppose we pad $x(n)$ with L zeros and compute the $(N + L)$-point DFT, $X'(k)$. What is the relationship between $X(0)$ and $X'(0)$? If we plot $|X(k)|$ and $|X'(k)|$ on the same graph, explain the relationships between the two graphs.

11.5 Show that the sequence $\{x_n\}$ given by Equation 11.2–11 corresponds to the samples of the signal $x(t)$ given by Equation 11.2–12.

11.6 Show that the IDFT of a sequence $\{X_k, 0 \leq k \leq N - 1\}$ can be computed by passing the sequence $\{X_k\}$ through a bank of N linear discrete-time filters with system functions

$$H_n(z) = \frac{1}{1 - e^{j2\pi n/N}z^{-1}}$$

and sampling the filter outputs at $n = N$.

11.7 Plot $P_2(L)$, given by Equation 11.1–24 for $L = 1$ and $L = 2$ as a function of $10 \log \gamma_b$ and determine the loss in SNR due to the combining loss for $\gamma_b = 10$.

12

Spread Spectrum Signals for Digital Communications

Spread spectrum signals used for the transmission of digital information are distinguished by the characteristic that their bandwidth W is much greater than the information rate R in bits/s. That is, the bandwidth expansion factor $B_e = W/R$ for a spread spectrum signal is much greater than unity. The large redundancy inherent in spread spectrum signals is required to overcome the severe levels of interference that are encountered in the transmission of digital information over some radio and satellite channels. Since coded waveforms are also characterized by a bandwidth expansion factor greater than unity and since coding is an efficient method for introducing redundancy, it follows that coding is an important element in the design of spread spectrum signals and systems.

A second important element employed in the design of spread spectrum signals is pseudorandomness, which makes the signals appear similar to random noise and difficult to demodulate by receivers other than the intended ones. This element is intimately related with the application or purpose of such signals.

To be specific, spread spectrum signals are used for

- Combating or suppressing the detrimental effects of interference due to jamming, interference arising from other users of the channel, and self-interference due to multipath propagation.
- Hiding a signal by transmitting it at low power and, thus, making it difficult for an unintended listener to detect in the presence of background noise.
- Achieving message privacy in the presence of other listeners.

In applications other than communications, spread spectrum signals are used to obtain accurate range (time delay) and range rate (velocity) measurements in radar and navigation. For the sake of brevity, we shall limit our discussion to digital communication applications.

In combating intentional interference (jamming), it is important to the communicators that the jammer who is trying to disrupt the communication does not have prior knowledge of the signal characteristics except for the overall channel bandwidth and the type of modulation (PSK, FSK, etc.) being used. If the digital information is just encoded as described in Chapters 7 and 8, a sophisticated jammer can easily mimic the signal emitted by the transmitter and, thus, confuse the receiver. To circumvent this possibility, the transmitter introduces an element of unpredictability or randomness (pseudorandomness) in each of the transmitted coded signal waveforms that is known to the intended receiver but not to the jammer. As a consequence, the jammer must synthesize and transmit an interfering signal without knowledge of the pseudorandom pattern.

Interference from the other users arises in multiple-access communication systems in which a number of users share a common channel bandwidth. At any given time, a subset of these users may transmit information simultaneously over the common channel to corresponding receivers. Assuming that all the users employ the same code for the encoding and decoding of their respective information sequences, the transmitted signals in this common spectrum may be distinguished from one another by superimposing a different pseudorandom pattern, also called a *code*, in each transmitted signal. Thus, a particular receiver can recover the transmitted information intended for it by knowing the pseudorandom pattern, i.e., the key, used by the corresponding transmitter. This type of communication technique, which allows multiple users to simultaneously use a common channel for transmission of information, is called *code division multiple access* (CDMA). CDMA will be considered in Sections 12.2 and 12.3.

Resolvable multipath components resulting from time-dispersive propagation through a channel may be viewed as a form of self-interference. This type of interference may also be suppressed by the introduction of a pseudorandom pattern in the transmitted signal, as will be described below.

A message may be hidden in the background noise by spreading its bandwidth with coding and transmitting the resultant signal at a low average power. Because of its low power level, the transmitted signal is said to be "covert." It has a low probability of being intercepted (detected) by a casual listener and, hence, is also called a *low-probability-of-intercept* (LPI) signal.

Finally, message privacy may be obtained by superimposing a pseudorandom pattern on a transmitted message. The message can be demodulated by the intended receivers, who know the pseudorandom pattern or key used at the transmitter, but not by any other receivers who do not have knowledge of the key.

In the following sections, we shall describe a number of different types of spread spectrum signals, their characteristics, and their applications. The emphasis will be on the use of spread spectrum signals for combating interference (antijam or AJ signals), CDMA, and LPI. Before discussing the signal design problem, however, we shall briefly describe the types of channel characteristics assumed for the applications cited above.

12.1

MODEL OF SPREAD SPECTRUM DIGITAL COMMUNICATION SYSTEM

The block diagram shown in Figure 12.1–1 illustrates the basic elements of a spread spectrum digital communication system with a binary information sequence at its input at the transmitting end and at its output at the receiving end. The channel encoder and decoder and the modulator and demodulator are basic elements of the system,

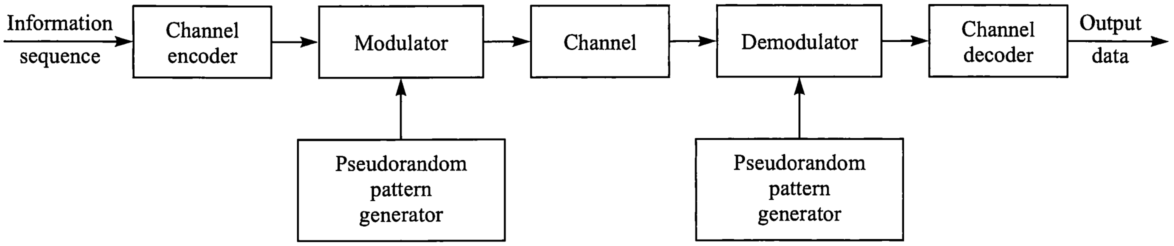


FIGURE 12.1–1
Model of spread spectrum digital communication system.

which were treated in Chapters 4, 7, and 8. In addition to these elements, we have two identical pseudorandom pattern generators, one that interfaces with the modulator at the transmitting end and a second that interfaces with the demodulator at the receiving end. The generators generate a pseudorandom or pseudonoise (PN) binary-valued sequence which is impressed on the transmitted signal at the modulator and removed from the received signal at the demodulator.

Synchronization of the PN sequence generated at the receiver with the PN sequence contained in the incoming received signal is required in order to demodulate the received signal. Initially, prior to the transmission of information, synchronization may be achieved by transmitting a fixed pseudorandom bit pattern that the receiver will recognize in the presence of interference with a high probability. After time synchronization of the generators is established, the transmission of information may commence.

Interference is introduced in the transmission of the information-bearing signal through the channel. The characteristics of the interference depend to a large extent on its origin. It may be categorized as being either broadband or narrowband relative to the bandwidth of the information-bearing signal and as either continuous or pulsed (discontinuous) in time. For example, an interfering signal may consist of one or more sinusoids in the bandwidth used to transmit the information. The frequencies of the sinusoids may remain fixed or they may change with time according to some rule. As a second example, the interference generated in CDMA by other users of the channel may be either broadband or narrowband, depending on the type of spread spectrum signal that is employed to achieve multiple access. If it is broadband, it may be characterized as an equivalent additive white Gaussian noise. We shall consider these types of interference and some others in the following sections.

Our treatment of spread spectrum signals will focus on the performance of the digital communication system in the presence of narrowband and broadband interference. Two types of modulation are considered: PSK and FSK. PSK is appropriate in applications where phase coherence between the transmitted signal and the received signal can be maintained over a time interval that is relatively long compared to the reciprocal of the transmitted signal bandwidth. On the other hand, FSK modulation is appropriate in applications where such phase coherence cannot be maintained due to time-variant effects on the communications link. This may be the case in a communications link between two high-speed aircraft or between a high-speed aircraft and a ground terminal.

The PN sequence generated at the modulator is used in conjunction with the PSK modulation to shift the phase of the PSK signal pseudorandomly as described in Section 12.2. The resulting modulated signal is called a *direct sequence* (DS) or a *pseudo-noise* (PN) spread spectrum signal. When used in conjunction with binary or M-ary ($M > 2$) FSK, the pseudorandom sequence selects the frequency of the transmitted signal pseudorandomly. The resulting signal is called a *frequency-hopped* (FH) spread spectrum signal. Although a number of other types of spread spectrum signals will be briefly described, the emphasis of our treatment will be on DS and FH spread spectrum signals.

12.2
DIRECT SEQUENCE SPREAD SPECTRUM SIGNALS

In the model shown in Figure 12.1–1, we assume that the information rate at the input to the encoder is R bits/s and the available channel bandwidth is W Hz. The modulation is assumed to be binary PSK. In order to utilize the entire available channel bandwidth, the phase of the carrier is shifted pseudorandomly according to the pattern from the PN generator at a rate W times/s. The reciprocal of W, denoted by T_c, defines the duration of a pulse, which is called a *chip*; T_c is called the *chip interval*. The pulse is the basic element in a DS spread spectrum signal.

If we define $T_b = 1/R$ to be the duration of a rectangular pulse corresponding to the transmission time of an information bit, the bandwidth expansion factor W/R may be expressed as

$$B_e = \frac{W}{R} = \frac{T_b}{T_c} \tag{12.2–1}$$

In practical systems, the ratio T_b/T_c is an integer,

$$L_c = \frac{T_b}{T_c} \tag{12.2–2}$$

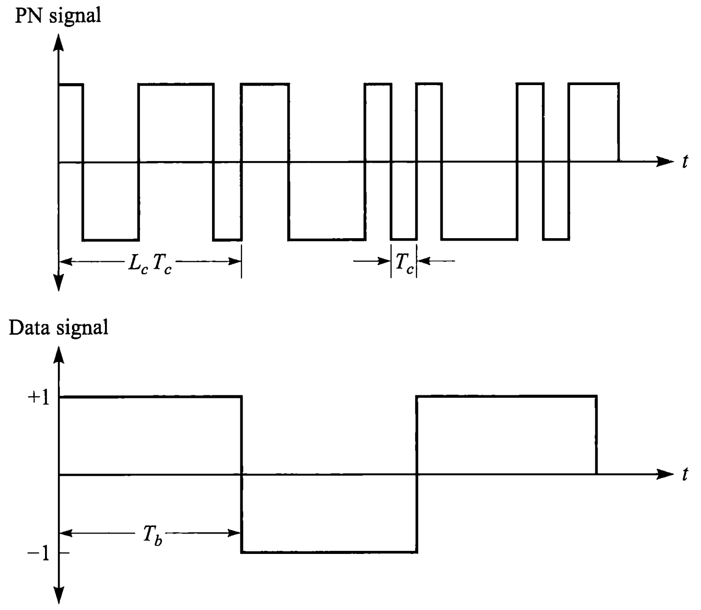
which is the number of chips per information bit. That is, L_c is the number of phase shifts that can occur in the transmitted signal during the bit duration $T_b = 1/R$. Figure 12.2–1a illustrates the relationships between the PN signal and the data signal.

Suppose that the encoder takes k information bits at a time and generates a binary linear (n, k) block code. The time duration available for transmitting the n code elements is kT_b seconds. The number of chips that occur in this time interval is kL_c. Hence, we may select the block length of the code as $n = kL_c$. If the encoder generates a binary convolutional code of rate k/n, the number of chips in the time interval kT_b is also $n = kL_c$. Therefore, the following discussion applies to both block codes and convolutional codes. We note that the code rate $R_c = k/n = 1/L_c$.

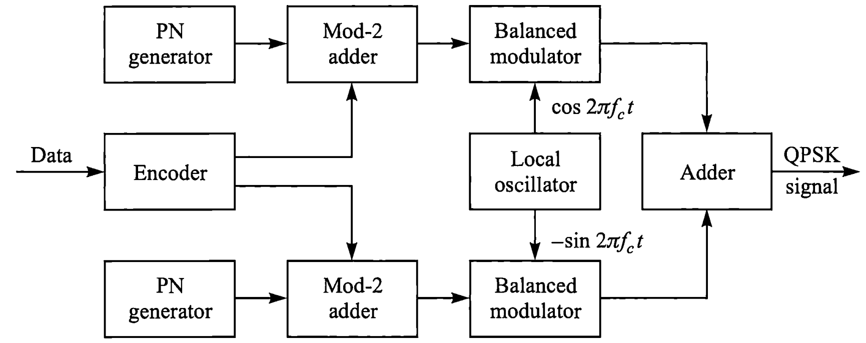
One method for impressing the PN sequence on the transmitted signal is to alter directly the coded bits by modulo-2 addition with the PN sequence.† Thus, each coded

---

†When four-phase PSK is desired, one PN sequence is added to the information sequence carried on the in-phase signal component and a second PN sequence is added to the information sequence carried on the quadrature component. In many PN spread spectrum systems, the same binary information sequence is added to the two PN sequences to form the two quadrature components. Thus, a four-phase PSK signal is generated with a binary information stream.



(a) PN and data signals


PN generator
Mod-2 adder
Balanced modulator
cos $2\pi f_c t$
Data
Encoder
Local oscillator
Adder
QPSK signal
$-\sin 2\pi f_c t$
PN generator
Mod-2 adder
Balanced modulator

(b) DS-QPSK modulator

FIGURE 12.2–1
The PN and data signals (a) and the QPSK modulator (b) for a DS spread spectrum system.

bit is altered by its addition with a bit from the PN sequence. If b_i represents the ith bit of the PN sequence and c_i is the corresponding bit from the encoder, the modulo-2 sum is

$$a_i = b_i \oplus c_i \tag{12.2–3}$$

Hence, $a_i = 1$ if either $b_i = 1$ and $c_i = 0$ or $b_i = 0$ and $c_i = 1$; also $a_i = 0$ if either $b_i = 1$ and $c_i = 1$ or $b_i = 0$ and $c_i = 0$. We may say that $a_i = 0$ when $b_i = c_i$ and $a_i = 1$ when $b_i \neq c_i$. The sequence $\{a_i\}$ is mapped into a binary PSK signal of the form $s(t) = \pm\text{Re}[g(t)e^{j2\pi f_c t}]$ according to the convention

$$g_i(t) = \begin{cases} g(t - iT_c) & a_i = 0 \\ -g(t - iT_c) & a_i = 1 \end{cases} \tag{12.2–4}$$

where $g(t)$ represents a pulse of duration T_c seconds and arbitrary shape.

The modulo-2 addition of the coded sequence $\{c_i\}$ and the sequence $\{b_i\}$ from the PN generator may also be represented as a multiplication of two waveforms. To demonstrate this point, suppose that the elements of the coded sequence are mapped into a binary PSK signal according to the relation

$$c_i(t) = (2c_i - 1)g(t - iT_c) \tag{12.2–5}$$

Similarly, we define a waveform $p_i(t)$ as

$$p_i(t) = (2b_i - 1)p(t - iT_c) \tag{12.2–6}$$

where $p(t)$ is a rectangular pulse of duration T_c. Then the equivalent low-pass transmitted signal corresponding to the ith coded bit is

$$\begin{aligned} g_i(t) &= p_i(t)\, c_i(t) \\ &= (2b_i - 1)(2c_i - 1)g(t - iT_c) \end{aligned} \tag{12.2–7}$$

This signal is identical to the one given by Equation 12.2–4, which is obtained from the sequence $\{a_i\}$. Consequently, modulo-2 addition of the coded bits with the PN sequence followed by a mapping that yields a binary PSK signal is equivalent to multiplying a binary PSK signal generated from the coded bits with a sequence of unit amplitude rectangular pulses, each of duration T_c, and with a polarity which is determined from the PN sequence according to Equation 12.2–6. Although it is easier to implement modulo-2 addition followed by PSK modulation instead of waveform multiplication, it is convenient, for purposes of demodulation, to consider the transmitted signal in the multiplicative form given by Equation 12.2–7. A functional block diagram of a four-phase PSK-DS spread spectrum modulator is shown in Figure 12.2–1(b).

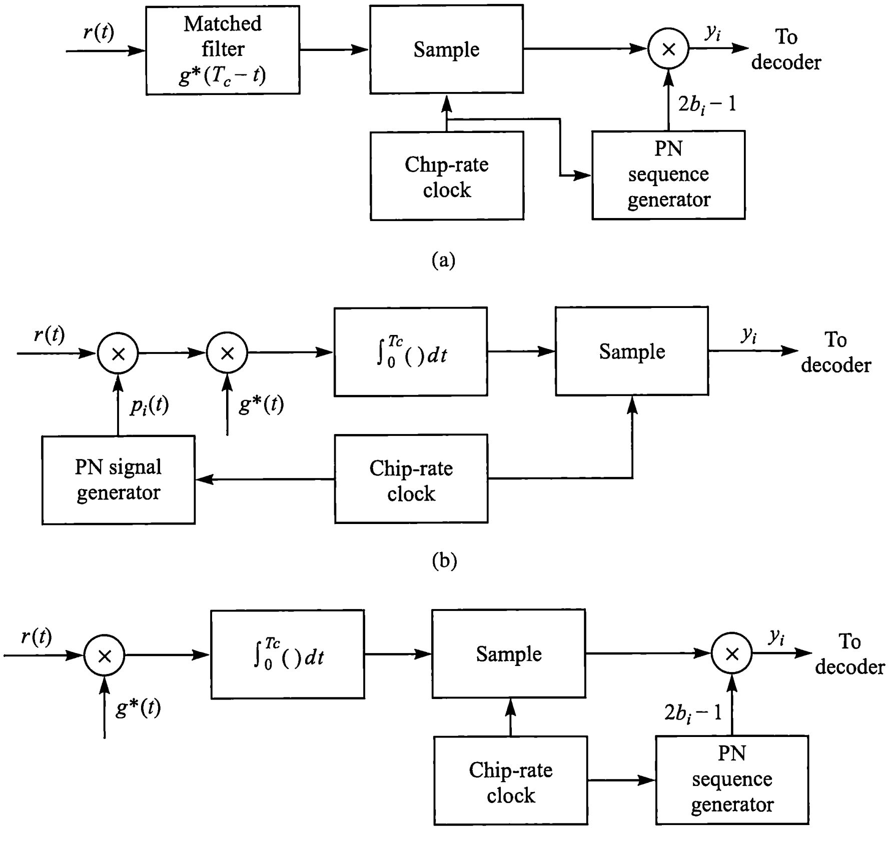
The received equivalent low-pass signal for the ith code element is

$$\begin{aligned} r_i(t) &= p_i(t)c_i(t) + z(t), \qquad iT_c \le t \le (i+1)T_c \\ &= (2b_i - 1)(2c_i - 1)g(t - iT_c) + z(t) \end{aligned} \tag{12.2–8}$$

where $z(t)$ represents the low-pass equivalent noise and interference signal corrupting the information-bearing signal. This signal is assumed to be a stationary random process with zero mean.

If $z(t)$ is a sample function from a complex-valued Gaussian process, the optimum demodulator may be implemented either as a filter matched to the waveform $g(t)$ or as a correlator, as illustrated by the block diagrams in Figure 12.2–2. In the matched filter realization, the sampled output from the matched filter is multiplied by $2b_i - 1$, which is obtained from the PN generator at the demodulator when the PN generator is properly synchronized. Since $(2b_i - 1)^2 = 1$ when $b_i = 0$ and $b_i = 1$, the effect of the PN sequence on the received coded bits is thus removed.

In Figure 12.2–2, we also observe that the cross correlation can be accomplished in either one of two ways. The first, illustrated in Figure 12.2–2b, involves premultiplying $r_i(t)$ with the waveform $p_i(t)$ generated from the output of the PN generator and then cross-correlating with $g^*(t)$ and sampling the output in each chip interval. The second method, illustrated in Figure 12.2–2c, involves cross correlation with $g^*(t)$ first, sampling the output of the correlator and, then, multiplying this output with $2b_i - 1$, which is obtained from the PN generator.


r(t)
Matched filter g*(T_c − t)
Sample
y_i
To decoder
2b_i − 1
Chip-rate clock
PN sequence generator
r(t)
p_i(t)
g*(t)
∫_0^{Tc} ( )dt
Sample
y_i
To decoder
PN signal generator
Chip-rate clock
r(t)
g*(t)
∫_0^{Tc} ( )dt
Sample
y_i
To decoder
2b_i − 1
Chip-rate clock
PN sequence generator

(a)

(b)

(c)

FIGURE 12.2–2
Possible demodulator structures for PN spread spectrum signals.

If $z(t)$ is not a Gaussian random process, the demodulation methods illustrated in Figure 12.2–2 are no longer optimum. Nevertheless, we may still use any of these three demodulator structures to demodulate the received signal. When the statistical characteristics of the interference $z(t)$ are unknown a priori, this is certainly one possible approach. An alternative method, which is described later, utilizes an adaptive filter prior to the matched filter or correlator to whiten the interference. The rationale for this second method is also described later.

In Section 12.2–1, we derive the error rate performance of the DS spread spectrum system in the presence of wideband and narrowband interference. The derivations are based on the assumption that the demodulator is any of the three equivalent structures shown in Figure 12.2–2.

12.2–1 Error Rate Performance of the Decoder

Let the unquantized output of the demodulator be denoted by y_j, $1 \le j \le n$. First we consider a linear binary (n, k) block code and, without loss of generality, we assume that the all-zero code word is transmitted.

A decoder that employs soft-decision decoding computes the correlation metrics

$$CM_i = \sum_{j=1}^{n}(2c_{ij} - 1)y_j, \qquad i = 1, 2, \ldots, 2^k \tag{12.2–9}$$

where c_{ij} denotes the jth bit in the ith code word. The correlation metric corresponding to the all-zero code word is

$$\begin{aligned} CM_1 &= 2n\mathcal{E}_c + \sum_{j=1}^{n}(2c_{1j} - 1)(2b_j - 1)\nu_j \\ &= 2n\mathcal{E}_c - \sum_{j=1}^{n}(2b_j - 1)\nu_j \end{aligned} \tag{12.2–10}$$

where ν_j, $1 \le j \le n$, is the additive noise and interference term corrupting the jth coded bit and $\mathcal{E}_c$ is the chip energy. It is defined as

$$\nu_j = \text{Re}\left\{\int_0^{T_c} g^*(t)z[t + (j-1)T_c]\,dt\right\}, \qquad j = 1, 2, \ldots, n \tag{12.2–11}$$

Similarly, the correlation metric corresponding to code word $\boldsymbol{c}_m$ having weight w_m is

$$CM_m = 2\mathcal{E}_c n\left(1 - \frac{2w_m}{n}\right) + \sum_{j=1}^{n}(2c_{mj} - 1)(2b_j - 1)\nu_j \tag{12.2–12}$$

Following the procedure used in Section 7.4, we shall determine the probability that $CM_m > CM_1$. The difference between CM_1 and CM_m is

$$\begin{aligned} D &= CM_1 - CM_m \\ &= 4\mathcal{E}_c w_m - 2\sum_{j=1}^{n} c_{mj}(2b_j - 1)\nu_j \end{aligned} \tag{12.2–13}$$

Since the codeword $\boldsymbol{c}_m$ has weight w_m, there are w_m nonzero components in the summation of noise terms contained in Equation 12.2–13. We shall assume that the minimum distance of the code is sufficiently large that we can invoke the central limit theorem for the summation of noise components. This assumption is valid for DS spread spectrum signals that have a bandwidth expansion of 10 or more.† Thus, the summation of noise components is modeled as a Gaussian random variable. Since $E(2b_j - 1) = 0$ and $E(\nu_j) = 0$, the mean of the second term in Equation 12.2–13 is also zero.

The variance is

$$\sigma_m^2 = 4\sum_{j=1}^{n}\sum_{i=1}^{n} c_{mi}c_{mj}E[(2b_j - 1)(2b_i - 1)]E(\nu_i\nu_j) \tag{12.2–14}$$

---

†Typically, the bandwidth expansion factor in a spread spectrum signal is of the order of 10 to 100 and sometimes higher.

The sequence of binary digits from the PN generator are assumed to be uncorrelated. Hence

$$E[(2b_j - 1)(2b_i - 1)] = \delta_{ij} \tag{12.2–15}$$

and

$$\sigma_m^2 = 4w_m E(\nu^2) \tag{12.2–16}$$

where $E(\nu^2)$ is the second moment of any one element from the set $\{\nu_j\}$. This moment is easily evaluated to yield

$$\begin{aligned} E(\nu^2) &= \frac{1}{2}\int_0^{T_c}\int_0^{T_c} g^*(t)g(\tau)R_{zz}(t-\tau)\,dt\,d\tau \\ &= \frac{1}{2}\int_{-\infty}^{\infty} |G(f)|^2 \mathcal{S}_{zz}(f)\,df \end{aligned} \tag{12.2–17}$$

where $R_{zz}(\tau) = E[z^*(t)z(t+\tau)]$ is the autocorrelation function and $\mathcal{S}_{zz}(f)$ is the power spectral density of the interference $z(t)$.

We observe that when the interference is spectrally flat within the bandwidth† occupied by the transmitted signal, i.e.,

$$\mathcal{S}_{zz}(f) = 2J_0, \qquad |f| \le \tfrac{1}{2}W \tag{12.2–18}$$

the second moment in Equation 12.2–17 is $E(\nu^2) = 2\mathcal{E}_c J_0$, and, hence, the variance of the interference term in Equation 12.2–16 becomes

$$\sigma_m^2 = 8\mathcal{E}_c J_0 w_m \tag{12.2–19}$$

In this case, the probability that $D < 0$ is

$$P_2(m) = Q\left(\sqrt{\frac{2\mathcal{E}_c}{J_0}w_m}\right) \tag{12.2–20}$$

But the energy per coded bit $\mathcal{E}_c$ may be expressed in terms of the energy per information bit $\mathcal{E}_b$ as

$$\mathcal{E}_c = \frac{k}{n}\mathcal{E}_b = R_c\mathcal{E}_b \tag{12.2–21}$$

With his substitution, Equation 12.2–20 becomes

$$\begin{aligned} P_2(m) &= Q\left(\sqrt{\frac{2\mathcal{E}_b}{J_0}R_c w_m}\right) \\ &= Q\left(\sqrt{2\gamma_b R_c w_m}\right) \end{aligned} \tag{12.2–22}$$

---

†If the bandwidth of the bandpass channel is W, that of the equivalent low-pass channel is $\frac{1}{2}W$.

where $\gamma_b = \mathcal{E}_b / J_0$ is the SNR per information bit. Finally, the code word error probability may be upper-bounded by the union bound as

$$P_M \leq \sum_{m=2}^{M} Q(\sqrt{2\gamma_b R_c w_m}) \tag{12.2–23}$$

where $M = 2^k$. Note that this expression is identical to the probability of a code word error for soft-decision decoding of a linear binary block code in an AWGN channel.

Although we have considered a binary block code in the derivation given above, the procedure is similar for an (n, k) convolutional code. The result of such a derivation is the following upper bound on the equivalent bit error probability:

$$P_b \leq \frac{1}{k} \sum_{d=d_{\text{free}}}^{\infty} \beta_d Q(\sqrt{2\gamma_b R_c d}) \tag{12.2–24}$$

The set of coefficients $\{\beta_d\}$ is obtained from an expansion of the derivative of the transfer function $T(Y, Z)$, as described in Section 8.2–2.

Next, we consider a narrowband interference centered at the carrier (at DC for the equivalent low-pass signal). We may fix the total (average) interference power to $J_{\text{av}} = 2J_0 W$, where $2J_0$ is the value of the power spectral density of an equivalent wideband interference. The narrowband interference is characterized by the power spectral density

$$\mathcal{S}_{zz}(f) = \begin{cases} \dfrac{J_{\text{av}}}{W_1} & |f| \leq \frac{1}{2}W_1 \\ 0 & |f| > \frac{1}{2}W_1 \end{cases} \tag{12.2–25}$$

where $W \gg W_1$.

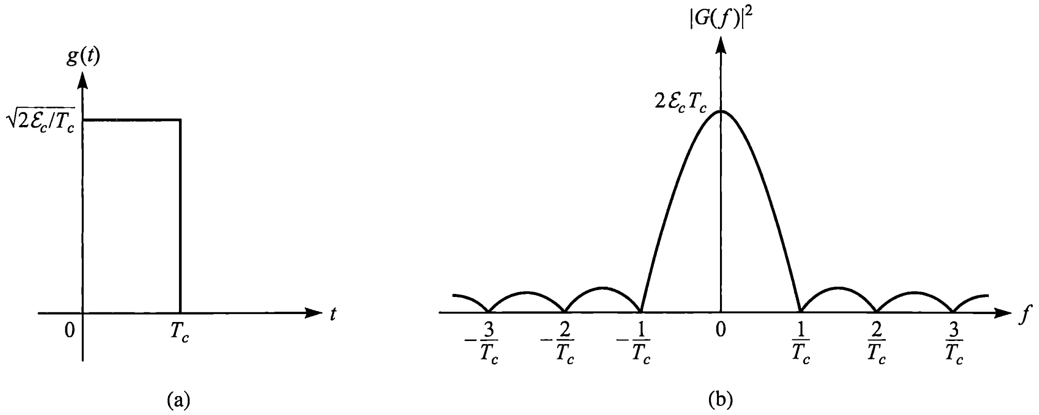
Substitution of Equation 12.2–25 for $\mathcal{S}_{zz}(f)$ into Equation 12.2–17 yields

$$E(\nu^2) = \frac{J_{\text{av}}}{2W_1} \int_{-W_1/2}^{W_1/2} |G(f)|^2 \, df \tag{12.2–26}$$

The value of $E(\nu^2)$ depends on the spectral characteristics of the pulse $g(t)$. In the following example, we consider two special cases.

EXAMPLE 12.2–1. Suppose that $g(t)$ is a rectangular pulse as shown in Figure 12.2–3(a) and $|G(f)|^2$ is the corresponding energy density spectrum shown in Figure 12.2–3(b). For the narrowband interference given by Equation 12.2–25, the variance of the total interference is

$$\begin{aligned} \sigma_m^2 &= 4w_m E(\nu^2) \\ &= \frac{4\mathcal{E}_c w_m T_c J_{\text{av}}}{W_1} \int_{-W_1/2}^{W_1/2} \left(\frac{\sin \pi f T_c}{\pi f T_c} \right)^2 df \\ &= \frac{4\mathcal{E}_c w_m J_{\text{av}}}{W_1} \int_{-\beta/2}^{\beta/2} \left(\frac{\sin \pi x}{\pi x} \right)^2 dx \end{aligned} \tag{12.2–27}$$



(a) (b)

FIGURE 12.2–3
Rectangular pulse and its energy density spectrum.

where $\beta = W_1 T_c$. Figure 12.2–4 illustrates the value of this integral for $0 \le \beta \le 1$. We observe that the value of the integral is upper-bounded by unity. Hence, $\sigma_m^2 \le 4\mathcal{E}_c w_m J_{\text{av}}/W_1$.

In the limit as W_1 becomes zero, the interference becomes an impulse at the carrier. In this case the interference is a pure frequency tone and it is usually called a *continuous wave (CW) interfering signal.* The power spectral density is

$$\mathcal{S}_{zz}(f) = J_{\text{av}}\delta(f) \tag{12.2–28}$$

and the corresponding variance for the decision variable $D = CM_1 - CM_m$ is

$$\begin{aligned} \sigma_m^2 &= 2w_m J_{\text{av}}|G(0)|^2 \\ &= 4w_m \mathcal{E}_c T_c J_{\text{av}} \end{aligned} \tag{12.2–29}$$

The probability of a codeword error for CW interference is upper-bounded as

$$P_e \le \sum_{m=2}^{M} Q\left(\sqrt{\frac{4\mathcal{E}_c}{J_{\text{av}} T_c}} w_m\right) \tag{12.2–30}$$

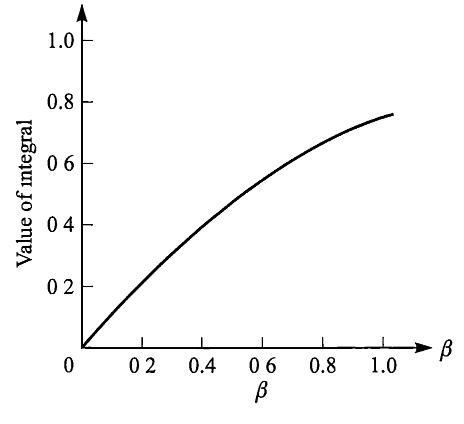


FIGURE 12.2–4
Plot of the value of the integral in Equation 12.2–27.

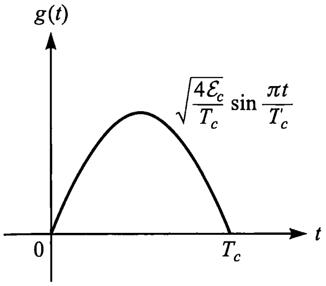


FIGURE 12.2–5
A sinusoidal signal pulse.

But $\mathcal{E}_c = R_c\mathcal{E}_b$. Furthermore, $T_c \approx 1/W$ and $J_{\text{av}}/W = 2J_0$. Therefore Equation 12.2–30 may be expressed as

$$P_e \leq \sum_{m=2}^{M} Q\left(\sqrt{\frac{2\mathcal{E}_b}{J_0}R_c w_m}\right) \tag{12.2–31}$$

which is the result obtained previously for broadband interference. This result indicates that a CW interference has the same effect on performance as an equivalent broadband interference. This equivalence is discussed further below.

EXAMPLE 12.2–2. Let us determine the performance of the DS spread spectrum system in the presence of a CW interference of average power J_{av} when the transmitted signal pulse $g(t)$ is one-half cycle of a sinusoid as illustrated in Figure 12.2–5, i.e.,

$$g(t) = \sqrt{\frac{4\mathcal{E}_c}{T_c}}\sin\frac{\pi t}{T_c}, \qquad 0 \leq t \leq T_c \tag{12.2–32}$$

The variance of the interference of this pulse is

$$\begin{aligned}\sigma_m^2 &= 2w_m J_{\text{av}}|G(0)|^2 \\ &= \frac{32}{\pi^2}\mathcal{E}_c T_c J_{\text{av}} w_m\end{aligned} \tag{12.2–33}$$

Hence, the upper bound on the codeword probability is

$$P_e \leq \sum_{m=2}^{M} Q\left(\sqrt{\frac{\pi^2\mathcal{E}_b}{2J_{\text{av}}T_c}R_c w_m}\right) \tag{12.2–34}$$

We observe that the performance obtained with this pulse is 0.9 dB better than that obtained with a rectangular pulse. Recall that this pulse shape when used in offset QPSK results in an MSK signal. MSK modulation is frequently used in DS spread spectrum systems.

The processing gain and the interference margin An interesting interpretation of the performance characteristics for the DS spread spectrum signal is obtained by expressing the signal energy per bit $\mathcal{E}_b$ in terms of the average power. That is, $\mathcal{E}_b = P_{\text{av}}T_b$, where P_{av} is the average signal power and T_b is the bit interval. Let us consider the performance obtained in the presence of CW interference for the rectangular pulse treated in Example 12.2–1. When we substitute for $\mathcal{E}_b$ and J_0 into Equation 12.2–31, we obtain

$$P_e \le \sum_{m=2}^{M} Q\left(\sqrt{\frac{4P_{\text{av}}}{J_{\text{av}}}\frac{T_b}{T_c}R_c w_m}\right) = \sum_{m=2}^{M} Q\left(\sqrt{\frac{4P_{\text{av}}}{J_{\text{av}}}L_c R_c w_m}\right) \tag{12.2–35}$$

where L_c is the number of chips per information bit and $P_{\text{av}}/J_{\text{av}}$ is the signal-to-interference power ratio.

An identical result is obtained with broadband interference for which the performance is given by Equation 12.2–23. For the signal energy per bit, we have

$$\mathcal{E}_b = P_{\text{av}}T_b = \frac{P_{\text{av}}}{R} \tag{12.2–36}$$

where R is the information rate in bits/s. The power spectral density for the interference may be expressed as

$$2J_0 = \frac{J_{\text{av}}}{W}$$

Using this relation and Equation 12.2–36, the ratio $\mathcal{E}_b/J_0$ may be expressed as

$$\frac{\mathcal{E}_b}{J_0} = \frac{P_{\text{av}}/R}{J_{\text{av}}/2W} = \frac{2W/R}{J_{\text{av}}/P_{\text{av}}} \tag{12.2–37}$$

The ratio $J_{\text{av}}/P_{\text{av}}$ is the interference-to-signal power ratio, which is usually greater than unity. The ratio $W/R = T_b/T_c = B_e = L_c$ is just the bandwidth expansion factor, or, equivalently, the number of chips per information bit. This ratio is usually called the *processing gain* of the DS spread spectrum system. It represents the advantage gained over the interference that is obtained by expanding the bandwidth of the transmitted signal. If we interpret $\mathcal{E}_b/J_0$ as the SNR required to achieve a specified error rate performanace and W/R as the available bandwidth expansion factor, the ratio $J_{\text{av}}/P_{\text{av}}$ is called the *interference margin* of the DS spread spectrum system. In other words, the interference margin is the largest value that the ratio $J_{\text{av}}/P_{\text{av}}$ can take and still satisfy the specified error probability.

The performance of a soft-decision decoder for a linear (n, k) binary code, expressed in terms of the processing gain and the interference margin, is

$$P_e \le \sum_{m=2}^{M} Q\left(\sqrt{\frac{4W/R}{J_{\text{av}}/P_{\text{av}}}R_c w_m}\right) \le (M-1)Q\left(\sqrt{\frac{4W/R}{J_{\text{av}}/P_{\text{av}}}R_c d_{\min}}\right) \tag{12.2–38}$$

In addition to the processing gain W/R and $J_{\text{av}}/P_{\text{av}}$, we observe that the performance depends on a third factor, namely, $R_c w_m$. This factor is the *coding gain.* A lower bound on this factor is $R_c d_{\min}$. Thus the interference margin achieved by the DS spread spectrum signal depends on the processing gain and the coding gain.

We may express the relationship among these three quantities in dB as

$$(\text{SNR})_{\text{dB}} = \left(\frac{2W}{R}\right)_{\text{dB}} + (R_c d_{\min})_{\text{dB}} - \left(\frac{J_{\text{av}}}{P_{\text{av}}}\right)_{\text{dB}} \tag{12.2–39}$$

where the $(\text{SNR})_{\text{dB}}$ is the signal-to-noise ratio required by the receiver to achieve a specified level of performance.

Uncoded DS spread spectrum signals The performance results given above for DS spread spectrum signals generated by means of an (n, k) code may be specialized to a trivial type of code, namely, a binary repetition code. For this case, $k = 1$ and the weight of the nonzero code word is $w = n$. Thus, $R_c w = 1$ and, hence, the performance of the binary signaling system reduces to

$$\begin{aligned} P_2 &= Q\left(\sqrt{\frac{2\mathcal{E}_b}{J_0}}\right) \\ &= Q\left(\sqrt{\frac{4W/R}{J_{\text{av}}/P_{\text{av}}}}\right) \end{aligned} \tag{12.2–40}$$

Note that the trivial (repetition) code gives no coding gain. It does result in a processing gain of W/R.

EXAMPLE 12.2–3. Suppose that we wish to achieve an error rate performance of 10^{-6} or less with an uncoded DS spread spectrum system. The available bandwidth expansion factor is $W/R = 1000$. Let us determine the jamming margin.

The $\mathcal{E}_b/J_0$ required to achieve a bit error probability of 10^{-6} with uncoded binary PSK is 10.5 dB. The processing gain is $10 \log_{10} 1000 = 30$ dB. Hence the maximum interference-to-signal power that can be tolerated, i.e., the interference margin, is

$$10 \log_{10} \frac{J_{\text{av}}}{P_{\text{av}}} = 33 - 10.5 = 22.5 \,\text{dB}$$

Since this is the interference margin achieved with an uncoded DS spread spectrum system, it may be increased by coding the information sequence.

There is another way to view the modulation and demodulation processes for the uncoded (repetition code) DS spread spectrum system. At the modulator, the signal waveform generated by the repetition code with rectangular pulses, for example, is identical to a unit amplitude rectangular pulse $s(t)$ of duration T_b or its negative, depending on whether the information bit is 1 or 0, respectively. This may be seen from Equation 12.2–7, where the coded chips $\{c_i\}$ within a single information bit are either all 1s or 0s. The PN sequence multiplies either $s(t)$ or $-s(t)$. Thus, when the information bit is a 1, the L_c PN chips generated by the PN generator are transmitted with the same polarity. On the other hand, when the information bit is a 0, the L_c PN chips when multiplied by $-s(t)$ are reversed in polarity.

The demodulator for the repetition code, implemented as a correlator, is illustrated in Figure 12.2–6. We observe that the integration interval in the integrator is the bit interval T_b. Thus, the decoder for the repetition code is eliminated and its function is subsumed in the demodulator.

Now let us qualitatively assess the effect of this demodulation process on the interference $z(t)$. The multiplication of $z(t)$ by the output of the PN generator, which is expressed as

$$w(t) = \sum_i (2b_i - 1)p(t - iT_c)$$

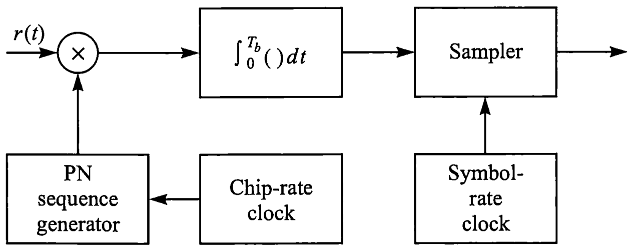


FIGURE 12.2–6
Correlation-type demodulator for a repetition code.

yields

$$v(t) = w(t)z(t)$$

The waveforms $w(t)$ and $z(t)$ are statistically independent random processes each with zero-mean and autocorrelation functions $R_{ww}(\tau)$ and $R_{zz}(\tau)$, respectively. The product $v(t)$ is also a random process having an autocorrelation function equal to the product of $R_{ww}(\tau)$ with $R_{zz}(\tau)$. Hence, the power spectral density of the process $v(t)$ is equal to the convolution of the power spectral density of $w(t)$ with the power spectral density of $z(t)$.

The effect of convolving the two spectra is to spread the power in bandwidth. Since the bandwidth of $w(t)$ occupies the available channel bandwidth W, the result of convolution of the two spectra is to spread the power spectral density of $z(t)$ over the frequency band of width W. If $z(t)$ is a narrowband process, i.e., its power spectral density has a width much less than W, the power spectral density of the process $v(t)$ will occupy a bandwidth equal to at least W.

The integrator used in the cross correlation shown in Figure 12.2–6 has a bandwidth approximately equal to $1/T_b$. Since $1/T_b \ll W$, only a fraction of the total interference power appears at the output of the correlator. This fraction is approximately equal to the ratio of bandwidths $1/T_b$ to W. That is,

$$\frac{1/T_b}{W} = \frac{1}{WT_b} = \frac{T_c}{T_b} = \frac{1}{L_c}$$

In other words, the multiplication of the interference with the signal from the PN generator spreads the interference to the signal bandwidth W, and the narrowband integration following the multiplication sees only the fraction $1/L_c$ of the total interference. Thus, the performance of the uncoded DS spread spectrum system is enhanced by the processing gain L_c.

Linear code concatenated with a repetition code As illustrated above, a binary repetition code provides a margin against an interference signal but yields no coding gain. To obtain an improvement in performance, we may use a linear (n_1, k) block or convolutional code, where $n_1 \le n = kL_c$. One possibility is to select $n_1 < n$ and to repeat each code bit n_2 times such that $n = n_1 n_2$. Thus, we can construct a linear (n, k) code by concatenating the (n_1, k) code with a binary $(n_2, 1)$ repetition code. This may be viewed as a trivial form of code concatenation where the outer code is the (n_1, k) code and the inner code is the repetition code.

Since the repetition code yields no coding gain, the coding gain achieved by the combined code must reduce to that achieved by the (n_1, k) outer code. It is demonstrated that this is indeed the case. The coding gain of the overall combined code is

$$R_c w_m = \frac{k}{n} w_m, \qquad m = 2, 3, \ldots, 2^k$$

But the weights $\{w_m\}$ for the combined code may be expressed as

$$w_m = n_2 w_m^o$$

where $\{w_m^o\}$ are the weights of the outer code. Therefore, the coding gain of the combined code is

$$R_c w_m = \frac{k}{n_1 n_2} n_2 w_m^o = \frac{k}{n_1} w_m^o = R_c^o w_m^o \tag{12.2–41}$$

which is just the coding gain obtained from the outer code.

A coding gain is also achieved if the (n_1, k) outer code is decoded using hard decisions. The probability of a bit error obtained with an $(n_2, 1)$ repetition code (based on soft-decision decoding) is

$$\begin{aligned} p &= Q\left(\sqrt{\frac{2n_2\mathcal{E}_c}{J_0}}\right) = Q\left(\sqrt{2\frac{\mathcal{E}_b}{J_0}R_c^o}\right) \\ &= Q\left(\sqrt{\frac{4W/R}{J_{\text{av}}/P_{\text{av}}}R_c^o}\right) \end{aligned} \tag{12.2–42}$$

Then the codeword error probability for a linear (n_1, k) block code is upper-bounded as

$$P_e \le \sum_{m=t+1}^{n_1} \binom{n_1}{m} p^m (1-p)^{n_1-m} \tag{12.2–43}$$

where $t = \lfloor \frac{1}{2}(d_{\min} - 1) \rfloor$, or as

$$P_e \le \sum_{m=2}^{M} [4p(1-p)]^{w_m^o/2} \tag{12.2–44}$$

where the latter is a Chernov bound. For an (n_1, k) binary convolutional code, the upper bound on the bit error probability is

$$P_b \le \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \tag{12.2–45}$$

where $P_2(d)$ is defined by Equation 8.2–16 for odd d and by Equation 8.2–17 for even d.

Concatenated coding for DS spread spectrum systems It is apparent from the above discussion that an improvement in performance can be obtained by replacing the repetition code by a more powerful code that will yield a coding gain in addition to the processing gain. Basically, the objective in a DS spread spectrum system is to construct a long, low-rate code having a large minimum distance. This may be best accomplished by using code concatenation. When binary PSK is used in conjunction with DS spread spectrum, the elements of a concatenated code word must be expressed in binary form.

Best performance is obtained when soft-decision decoding is used on both the inner and outer codes. However, an alternative, which usually results in reduced complexity for the decoder, is to employ soft-decision decoding on the inner code and hard-decision decoding on the outer code. The expressions for the error rate performance of these decoding schemes depend, in part, on the type of codes (block or convolutional) selected for the inner and outer codes. For example, the concatenation of two block codes may be viewed as an overall long binary (n, k) block code having a performance given by Equation 12.2–38. The performance of other code combinations may also be readily derived. For the sake of brevity, we shall not consider such code combinations.

12.2–2 Some Applications of DS Spread Spectrum Signals

In this subsection, we shall briefly consider the use of coded DS spread spectrum signals for two specific applications. One is concerned with a communication signal that is hidden in the background noise by transmitting the signal at a very low power level. The second application is concerned with accommodating a number of simultaneous signal transmissions on the same channel, i.e., CDMA.

Low-detectability signal transmission In this application, the signal is purposely transmitted at a very low power level relative to the background channel noise and thermal noise that is generated in the front end of the receiver. If the DS spread spectrum signal occupies a bandwidth W and the spectral density of the additive noise is $N_0/2$ W/Hz, the average noise power in the bandwidth W is $N_{\text{av}} = WN_0$.

The average received signal power at the intended receiver is P_{av}. If we wish to hide the presence of the signal from receivers that are in the vicinity of the intended receiver, the signal is transmitted at a low power level such that $P_{\text{av}}/N_{\text{av}} \ll 1$. For example, let us assume that binary PSK is used to transmit the information. The probability of error at the intended receiver may be expressed as

$$\begin{aligned} P_e &< MQ\left(\sqrt{\frac{2\mathcal{E}_b}{N_0}R_c d_{\min}}\right) \\ &< MQ\left(\sqrt{4\left(\frac{W}{R}\right)\left(\frac{P_{\text{av}}}{N_{\text{av}}}\right)R_c d_{\min}}\right) \end{aligned}$$

From this expression, we observe that even though $P_{\text{av}}/N_{\text{av}} \ll 1$, the intended receiver can recover the information-bearing signal with the aid of the processing gain and the coding gain. However, any other receiver that has no prior knowledge of the PN sequence is unable to take advantage of the processing gain and the coding gain. Hence, the presence of the information-bearing signal is difficult to detect. We say that the signal has a *low probability* of being *intercepted* (LPI) and it is called an *LPI signal.*

The probability of error results given in Section 12.2–1 also apply to the demodulation and decoding of LPI signals at the intended receiver.

Code division multiple access The enhancement in performance obtained from a DS spread spectrum signal through the processing gain and coding gain can be used to enable many DS spread spectrum signals to occupy the same channel bandwidth provided that each signal has its own distinct PN sequence. Thus, it is possible to have several users transmit messages simultaneously over the same channel bandwidth. This type of digital communication in which each user (transmitter–receiver pair) has a distinct PN code for transmitting over a common channel bandwidth is called *code division multiple access* (CDMA).

In the demodulation of each PN signal, the signals from the other simultaneous users of the channel appear as an additive interference. The level of interference varies, depending on the number of users at any given time. A major advantage of CDMA is that a large number of users can be accommodated if each transmits messages for a short period of time. In such a multiple access system, it is relatively easy either to add new users or to decrease the number of users without disrupting the system.

Let us determine the number of simultaneous signals that can be supported in a CDMA system.[†] For simplicity, we assume that all signals have identical average powers. Thus, if there are N_u simultaneous users, the desired signal-to-noise inteference power ratio at a given receiver is

$$\frac{P_{\text{av}}}{J_{\text{av}}} = \frac{P_{\text{av}}}{(N_u - 1)P_{\text{av}}} = \frac{1}{N_u - 1} \tag{12.2–46}$$

Hence, the performance for soft-decision decoding at the given receiver is upper-bounded as

$$P_e \leq \sum_{m=2}^{M} Q\left(\sqrt{\frac{4W/R}{N_u - 1} R_c w_m}\right) \leq (M-1)Q\left(\sqrt{\frac{4W/R}{N_u - 1} R_c d_{\min}}\right) \tag{12.2–47}$$

In this case, we have assumed that the interference from other users is Gaussian.

As an example, suppose that the desired level of performance (error probability of 10^{-6}) is achieved when

$$\frac{4W/R}{N_u - 1} R_c d_{\min} = 40$$

[†] In this section the interference from other users is treated as a random process. This is the case if there is no cooperation among the users. In Chapter 16 we consider CDMA transmission in which interference from other users is known and is suppressed by the receiver.

Then the maximum number of users that can be supported in the CDMA system is

$$N_u = \frac{W/R}{10} R_c d_{\min} + 1 \tag{12.2–48}$$

If $W/R = 100$ and $R_c d_{\min} = 4$, as obtained with the Golay (24, 12) code, the maximum number is $N_u = 41$. If $W/R = 1000$ and $R_c d_{\min} = 4$, this number becomes $N_u = 401$.

In determining the maximum number of simultaneous users of the channel, we have implicitly assumed that the PN code sequences are mutually orthogonal and the interference from other users adds on a power basis only. However, orthogonality among a number of PN code sequences is not easily achieved, especially if the number of PN code sequences required is large. In fact, the selection of a good set of PN sequences for a CDMA system is an important problem that has received considerable attention in the technical literature. We shall briefly discuss this problem in Section 12.2–5.

Digital cellular CDMA system based on DS spread spectrum Direct sequence CDMA has been adopted as one multiple-access method for digital cellular voice communications in North America. This digital cellular communication system was proposed and developed by Qualcomm and has been standardized and designated as IS-95 by the Telecommunications Industry Association (TIA) for use in the 800-MHz and in the 1900-MHz frequency bands.

The nominal bandwidth used for transmission from a base station to the mobile receivers (forward link) is 1.25 MHz, and a separate channel, also with a bandwidth of 1.25 MHz, is used for signal transmission from mobile receivers to a base station (reverse link). The signals transmitted in both the forward and the reverse links are DS spread spectrum signals having a chip rate of 1.2288×10^6 chips per second (Mchips/s).

Forward link A block diagram of the modulator for the signals transmitted from a base station to the mobile receivers is shown in Figure 12.2–7. The speech coder is a code-excited linear predictive (CELP) coder which generates data at the variable rates of 9600, 4800, 2400, and 1200 bits/s, where the data rate is a function of the speech activity of the user, in frame intervals of 20 ms. The data from the speech coder is encoded by a rate $1/2$, constraint length $K = 9$ convolutional code. For lower speech activity, where the data rates are 4800, 2400, or 1200 bits/s, the output symbols from the convolutional encoder are repeated either twice, four times, or eight times so as to maintain a constant bit rate of 9600 bits/s. At the lower speech activity rates, the transmitter power is reduced by either 3, 6, or 9 dB, so that the transmitted energy per bit remains constant for all speech rates. Thus, a lower speech activity results in a lower transmitter power and, hence, a lower level of interference to other users.

The encoded bits for each frame are passed through a block interleaver, which is needed to overcome the effects of burst errors that may occur in transmission through the channel. The data bits at the output of the block interleaver, which occur at a rate of 19.2 kbits/s, are scrambled by multiplication with the output of a long code (period $N = 2^{42} - 1$) generator running at the chip rate of 1.2288 M chips/s, but whose output is decimated by a factor of 64 to 19.2 kchips/s. The long code is used to uniquely identify a call of a mobile station on the forward and reverse links.

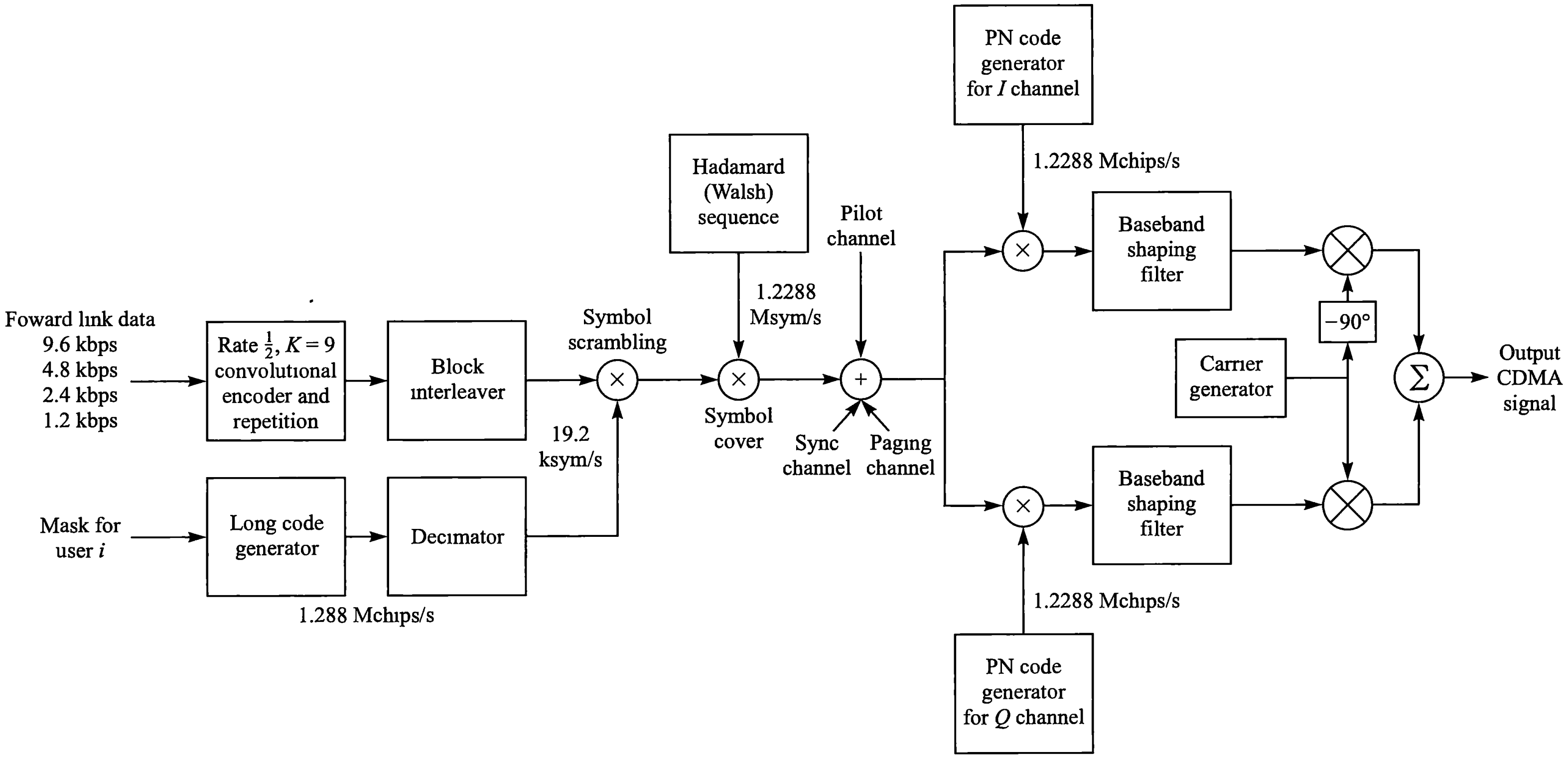

Foward link data
9.6 kbps
4.8 kbps
2.4 kbps
1.2 kbps
Rate $\frac{1}{2}$, $K = 9$ convolutional encoder and repetition
Block interleaver
Mask for user i
Long code generator
Decimator
1.288 Mchips/s
19.2 ksym/s
Symbol scrambling
Hadamard (Walsh) sequence
1.2288 Msym/s
Symbol cover
Pilot channel
Sync channel
Paging channel
PN code generator for I channel
1.2288 Mchips/s
PN code generator for Q channel
1.2288 Mchips/s
Baseband shaping filter
Baseband shaping filter
Carrier generator
−90°
Σ
Output CDMA signal

FIGURE 12.2–7
Block diagram of IS-95 forward link.

Each user of the channel is assigned a Hadamard (or Walsh) sequence of length 64. There are 64 orthogonal Hadamard sequences assigned to each base station, and, thus, there are 64 channels available. One Hadamard sequence (the all-zero sequence) is used to transmit a pilot signal, which serves as a means for measuring the channel characteristics, including the signal strength and the carrier phase offset. These parameters are used at the receiver in performing phase coherent demodulation. Another Hadamard sequence is used for providing time synchronization. One channel, and possibly more if necessary, is used for paging. That leaves up to 61 channels for allocation to different users.

Each user, using the Hadamard sequence assigned to it, multiplies the data sequence by the assigned Hadamard sequence. Thus, each encoded data bit is multiplied by the Hadamard sequence of length 64. The resulting binary sequence is now spread by multiplication with two PN sequences of length $N = 2^{15}$, so as to create in-phase and quadrature signal components. Thus, the binary data signal is converted to a four-phase signal and both the I and Q components are filtered by baseband spectral shaping filters. Different base stations are identified by different offsets of these PN sequences.The signals for all the 64 channels are transmitted synchronously so that, in the absence of channel multipath distortion, the signals of other users received at any mobile receiver do not interfere because of the orthogonality of the Hadamard sequences.

At the receiver, a RAKE demodulator is used to resolve the major multipath signal components, which are then phase-aligned and weighted according to their signal strength using the estimates of phase and signal strength derived from the pilot signal. These components are combined and passed to the Viterbi soft-decision decoder. The RAKE demodulator is described in detail in Chapter 13.

Reverse link The modulator for the reverse link from a mobile transmitter to a base station is different from that for the forward link. A block diagram of the modulator is shown in Figure 12.2–8. An important consideration in the design of the modulator is that signals transmitted from the various mobile transmitters to the base station are asynchronous and, hence, there is significantly more interference among users. Secondly, the mobile transmitters are usually battery operated and, consequently, these transmissions are power limited. To compensate for these major limitations, a $K = 9$, rate $1/3$ convolutional code is used in the reverse link. Although this code has essentially the same coding gain in an AWGN channel as the rate $1/2$ code used in the forward link, it has a much higher coding gain in a fading channel, which is the characteristic of digital cellular communication links, as we shall observe in our treatment of communication through fading channels in Chapter 13. As in the case of the forward link, for lower speech activity, the output bits from the convolutional encoder are repeated either two, or four, or eight times. However, the coded bit rate is 28.8 kbits/s.

For each 20-ms frame, the 576 encoded bits are block-interleaved and passed to the modulator. The data is modulated using an $M = 64$ orthogonal signal set using Hadamard sequences of length 64. Thus, a 6-bit block of data is mapped into one of the 64 Hadamard sequences. The result is a bit (or chip) rate of 307.2 kbits/s at the output of the modulator. We note that 64-ary orthogonal modulation at an error probability of 10^{-6} requires approximately 3.5 dB less SNR per bit than binary antipodal signaling.

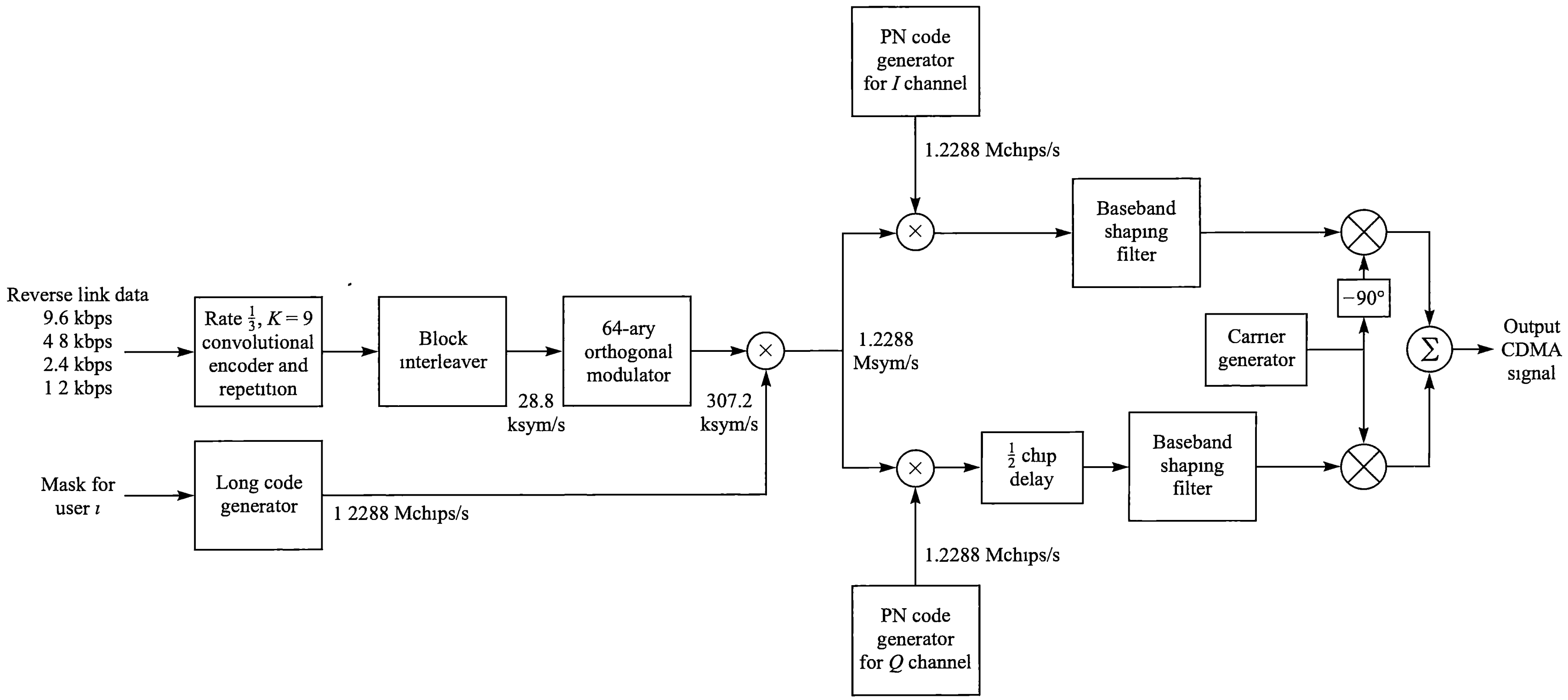

PN code generator for I channel
1.2288 Mchips/s
Baseband shaping filter
Reverse link data
9.6 kbps
4 8 kbps
2.4 kbps
1 2 kbps
Rate $\frac{1}{3}$, $K = 9$ convolutional encoder and repetition
Block interleaver
28.8 ksym/s
64-ary orthogonal modulator
307.2 ksym/s
1.2288 Msym/s
−90°
Carrier generator
Σ
Output CDMA signal
Mask for user i
Long code generator
1 2288 Mchips/s
$\frac{1}{2}$ chip delay
Baseband shaping filter
1.2288 Mchips/s
PN code generator for Q channel

FIGURE 12.2–8
Block diagram of IS-95 reverse link.

To reduce interference to other users, the time position of the transmitted code symbol repetitions is randomized so that, at the lower speech activity, consecutive bursts do not occur evenly spaced in time. Following the randomizer, the signal is spread by the output of the long code PN generator, which is running at a rate of 1.2288 Mchips/s. Hence, there are only four PN chips for every bit of the Hadamard sequence from the modulator, so the processing gain in the reverse link is very small. The resulting 1.2288 Mchips/s binary sequence at the output of the multiplier is then further multiplied by two PN sequences of length $N = 2^{15}$, whose rate is also 1.2288 Mchips/s, to create I and Q signals (a QPSK signal) which are filtered by baseband spectral shaping filters and then passed to quadrature mixers. The Q-channel signal is delayed in time by one-half PN chip relative to the I-channel signal prior to the baseband filter. In effect, the signal at the output of the two baseband filters is an offset QPSK signal.

Although the chips are transmitted as an offset QPSK signal, the demodulator employs noncoherent demodulation of the $M = 64$ orthogonal Hadamard waveforms to recover the encoded data bits. A fast Hadamard transform is used to reduce the computational complexity in the demodulation process. The output of the demodulator is then fed to the Viterbi detector, whose output is used to synthesize the speech signal.

12.2–3 Effect of Pulsed Interference on DS Spread Spectrum Systems

Thus far, we have considered the effect of continuous interference or jamming on a DS spread spectrum signal. We have observed that the processing gain and coding gain provide a means for overcoming the detrimental effects of this type of interference. However, there is a jamming threat that has a dramatic effect on the performance of a DS spread spectrum system. That jamming signal consists of pulses of spectrally flat noise that covers the entire signal bandwidth W. This is usually called *pulsed interference*.

Suppose the jammer has an average power J_{av} in the signal bandwidth W. Hence $2J_0 = J_{\text{av}}/W$. Instead of transmitting continuously, the jammer transmits pulses at a power J_{av}/α for α percent of the time, i.e., the probability that the jammer is transmitting at a given instant is α. For simplicity, we assume that an interference pulse spans an integral number of signaling intervals and, thus, it affects an integral number of bits. When the jammer is not transmitting, the transmitted bits are assumed to be received error-free, and when the jammer is transmitting, the probability of error for an uncoded DS spread spectrum system is $Q(\sqrt{2\alpha\mathcal{E}_b/J_0})$. Hence, the average probability of a bit error is

$$P_2(\alpha) = \alpha Q\left(\sqrt{2\alpha\mathcal{E}_b/J_0}\right) \tag{12.2–49}$$

The jammer selects the duty cycle α to maximize the error probability. On differentiating Equation 12.2–49 with respect to α, we find that the worst-case pulse jamming occurs when

$$\alpha^* = \begin{cases} \dfrac{0.71}{\mathcal{E}_b/J_0} & \mathcal{E}_b/J_0 \geq 0.71 \\ 1 & \mathcal{E}_b/J_0 < 0.71 \end{cases} \tag{12.2–50}$$

and the corresponding error probability is

$$P_2 = \begin{cases} \dfrac{0.083}{\mathcal{E}_b/J_0} & \mathcal{E}_b/J_0 > 0.71 \\ Q\left(\sqrt{\dfrac{2\mathcal{E}_b}{J_0}}\right) & \mathcal{E}_b/J_0 < 0.71 \end{cases} \tag{12.2–51}$$

The error rate performance given by Equation 12.2–49 for $\alpha = 1.0, 0.1$, and 0.01 along with the worst-case performance based on α^* is plotted in Figure 12.2–9. By comparing the error rate for continuous Gaussian noise jamming with worst-case pulse jamming, we observe a large difference in performance, which is approximately 40 dB at an error rate of 10^{-6}.

We should point out that the above analysis applies when the jammer pulse duration is equal to or greater than the bit duration. In addition, we should indicate that practical considerations may prohibit the jammer from achieving high peak power (small values of α). Nevertheless, the error probability given by Equation 12.2–51 serves as an upper bound on the performance of the uncoded binary PSK in worst-case pulse jamming. Clearly, the performance of the DS spread spectrum system in the presence of such interference is extremely poor.

If we simply add coding to the DS spread spectrum system, the improvement over the uncoded system is the coding gain. Thus, $\mathcal{E}_b/J_0$ is reduced by the coding gain,

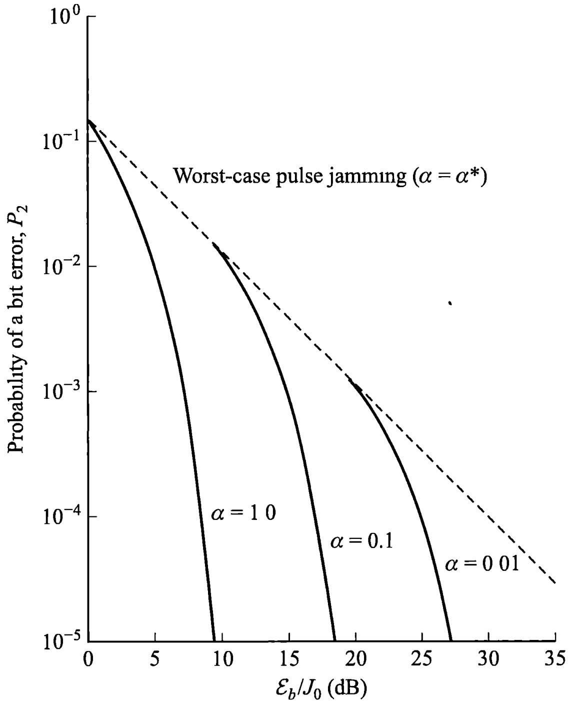


FIGURE 12.2–9
Performance of DS binary PSK with pulse interference.

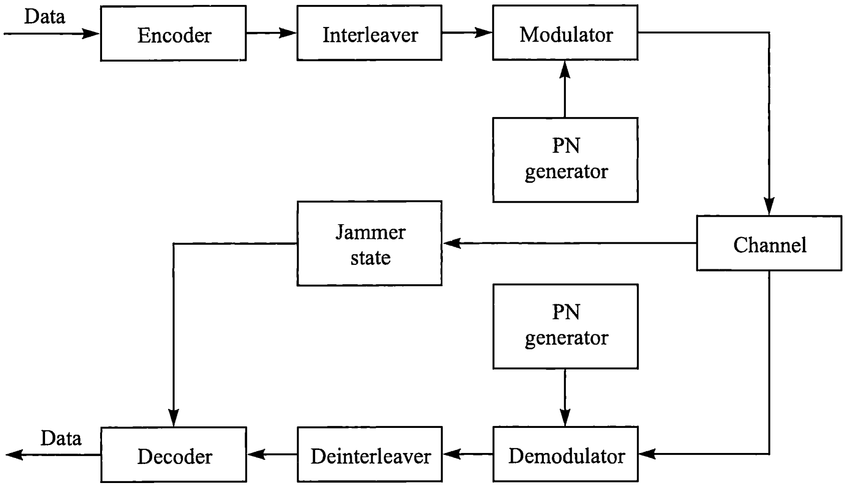


FIGURE 12.2–10
Block diagram of AJ communication system.

which in most cases is limited to less than 10 dB. The reason for the poor performance is that the jamming signal pulse duration may be selected to affect many consecutive coded bits when the jamming signal is turned on. Consequently, the code word error probability is high due to the burst characteristics of the jammer.

In order to improve the performance, we should interleave the coded bits prior to transmission over the channel. The effect of the interleaving, as discussed in Section 7.12, is to make the coded bits that are hit by the jammer statistically independent.

The block diagram of the digital communication system that includes interleaving/deinterleaving is shown in Figure 12.2–10. Also shown is the possibility that the receiver knows the jammer state, i.e., that it knows when the jammer is on or off. Knowledge of the jammer state (called *side information*) is sometimes available from channel measurements of noise power levels in adjacent frequency bands. In our treatment, we consider two extreme cases, namely, no knowledge of the jammer state or complete knowledge of the jammer state. In any case, the random variable ζ representing the jammer state is characterized by the probabilities

$$P(\zeta = 1) = \alpha, \qquad P(\zeta = 0) = 1 - \alpha \tag{12.2–52}$$

When the jammer is on, the channel is modeled as an AWGN with power spectral density $N_0 = J_0/\alpha$; and when the jammer is off, there is no noise in the channel. Knowledge of the jammer state implies that the decoder knows when $\zeta = 1$ and when $\zeta = 0$, and uses this information in the computation of the correlation metrics. For example, the decoder may weight the demodulator output for each coded bit by the reciprocal of the noise power level in the interval. Alternatively, the decoder may give zero weight (erasure) to a jammed bit.

First, let us consider the effect of jamming without knowledge of the jammer state. The interleaver/deinterleaver pair is assumed to result in statistically independent jammer hits of the coded bits. As an example of the performance achieved with coding, we cite the performance results from the paper of Martin and McAdam (1980). There the performance of binary convolutional codes is evaluated for worst-case pulse jamming. Both hard- and soft-decision Viterbi decoding are considered. Soft decisions

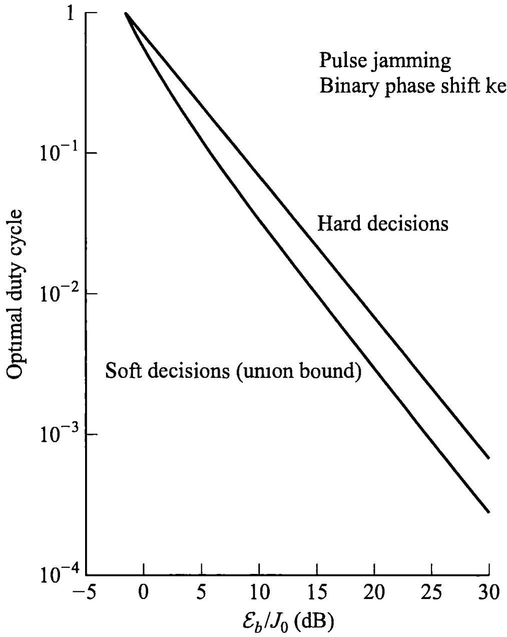


FIGURE 12.2–11
Optimal duty cycle for pulse jammer. [*From Martin and McAdam (1980).* © *1980 IEEE.*]

are obtained by quantizing the demodulator output to eight levels. For this purpose, a uniform quantizer is used for which the threshold spacing is optimized for the pulse jammer noise level. The quantizer plays the important role of limiting the size of the demodulator output when the pulse jammer is on. The limiting action ensures that any hit on a coded bit does not heavily bias the corresponding path metrics.

The optimum duty cycle for the pulse jammer in the coded system is generally inversely proportional to the SNR, but its value is different from that given by Equation 12.2–50 for the uncoded system. Figure 12.2–11 illustrates graphically the optimal jammer duty cycle for both hard- and soft-decision decoding of the rate 1/2 convolutional codes. The corresponding error rate results for this worst-case pulse jammer are illustrated in Figures 12.2–12 and 12.2–13 for rate 1/2 codes with constraint lengths $3 \leq K \leq 9$. For example, note that at $P_2 = 10^{-6}$, the $K = 7$ convolutional code with soft-decision decoding requires $\mathcal{E}_b/J_0 = 7.6$ dB, whereas hard-decision decoding requires $\mathcal{E}_b/J_0 = 11.7$ dB. This 4.1-dB difference in SNR is relatively large. With continuous Gaussian noise, the corresponding SNRs for an error rate of 10^{-6} are 5 dB for soft-decision decoding and 7 dB for hard-decision decoding. Hence, the worst-case pulse jammer has degraded the performance by 2.6 dB for soft-decision decoding and by 4.7 dB for hard-decision decoding. These levels of degradation increase as the constraint length of the convolutional code is decreased. The important point, however, is that the loss in SNR due to jamming has been reduced from 40 dB for the uncoded system to less than 5 dB for the coded system based on a $K = 7$, rate 1/2 convolutional code with interleaving.

A simpler method for evaluating the performance of a coded anti-jamming (AJ) communication system is to use the cutoff rate parameter R_0 as proposed by Omura and Levitt (1982). For example, with binary-coded modulation, the cutoff rate may be expressed as

$$R_0 = 1 - \log(1 + \Delta_\alpha) \qquad (12.2\text{–}53)$$

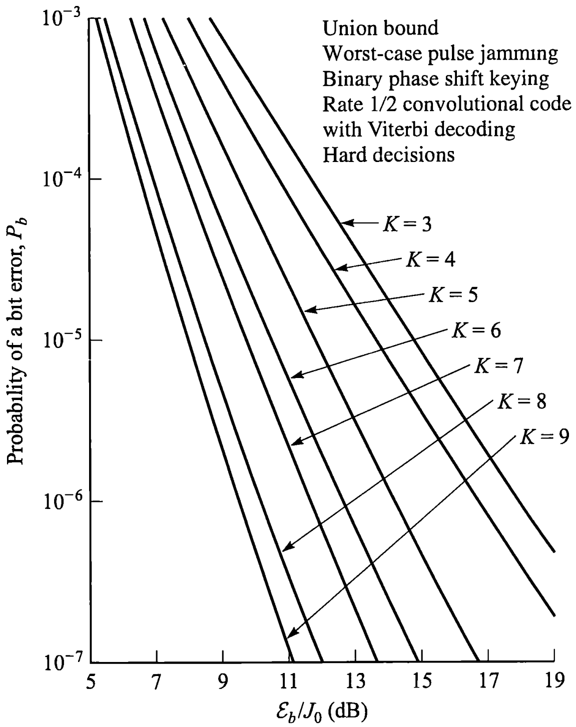


FIGURE 12.2–12
Performance of rate 1/2 convolutional codes with hard-decision Viterbi decoding binary PSK with worst-case pulse jamming. [*From Martin and McAdam (1980).* © *1980 IEEE.*]

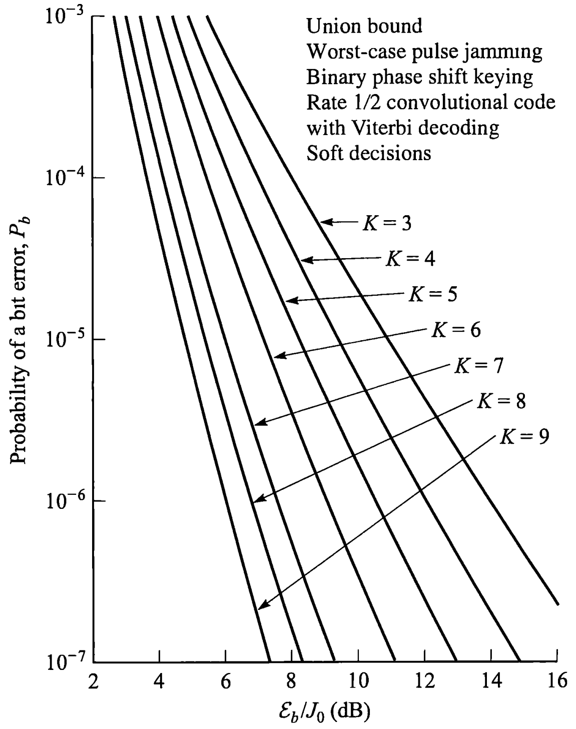


FIGURE 12.2–13
Performance of rate 1/2 convolutional codes with soft-decision Viterbi decoding binary PSK with worst-case pulse jamming. [*From Martin and McAdam (1980).* © *1980 IEEE.*]

where the factor Δ_α depends on the channel noise characteristics and the decoder processing. Recall that for binary PSK in an AWGN channel and soft-decision decoding,

$$\Delta_\alpha = e^{-\mathcal{E}_c/N_0} \tag{12.2–54}$$

where $\mathcal{E}_c$ is the energy per coded bit; and for hard-decision decoding,

$$\Delta_\alpha = \sqrt{4p(1-p)} \tag{12.2–55}$$

where p is the probability of a coded bit error. Here, we have $N_0 \equiv J_0$.

For a coded binary PSK, with pulse jamming, Omura and Levitt (1982) have shown that

$$\Delta_\alpha = \alpha e^{-\alpha\mathcal{E}_c/N_0} \qquad \text{for soft-decision decoding with knowledge of jammer state} \tag{12.2–56}$$

$$\Delta_\alpha = \min_{\lambda\geq 0}\left\{\left[\alpha\exp\left(\lambda^2\mathcal{E}_c/N_0/\alpha\right)+1-\alpha\right]\exp(-2\lambda\mathcal{E}_c)\right\} \qquad \text{for soft-decision decoding with no knowledge of jammer state} \tag{12.2–57}$$

$$\Delta_\alpha = \alpha\sqrt{4p(1-p)} \qquad \text{for hard-decision decoding with knowledge of the jammer state} \tag{12.2–58}$$

$$\Delta_\alpha = \sqrt{4\alpha p(1-\alpha p)} \qquad \text{for hard-decision decoding with no knowledge of the jammer state} \tag{12.2–59}$$

where the probability of error for hard-decision decoding of binary PSK is

$$p = Q\left(\sqrt{\frac{2\alpha\mathcal{E}_c}{N_0}}\right)$$

The graphs for R_0 as a function of $\mathcal{E}_c/N_0$ are illustrated in Figure 12.2–14 for the cases given above. Note that these graphs represent the cutoff rate for the worst-case value of $\alpha = \alpha^*$ that maximizes Δ_α (minimizes R_0) for each value of $\mathcal{E}_c/N_0$. Furthermore, note that with soft-decision decoding and no knowledge of the jammer state, $R_0 = 0$. This situation results from the fact that the demodulator output is not quantized.

The graphs in Figure 12.2–14 may be used to evaluate the performance of coded systems. To demonstrate the procedure, suppose that we wish to determine the SNR required to achieve an error probability of 10^{-6} with coded binary PSK in worst-case pulse jamming. To be specific, we assume that we have a rate $1/2$, $K = 7$ convolutional code. We begin with the performance of the rate $1/2$, $K = 7$ convolutional code with soft-decision decoding in an AWGN channel. At $P_2 = 10^{-6}$, the SNR required is found from Figure 8.6–1 to be

$$\frac{\mathcal{E}_b}{N_0} = 5 \text{ dB}$$

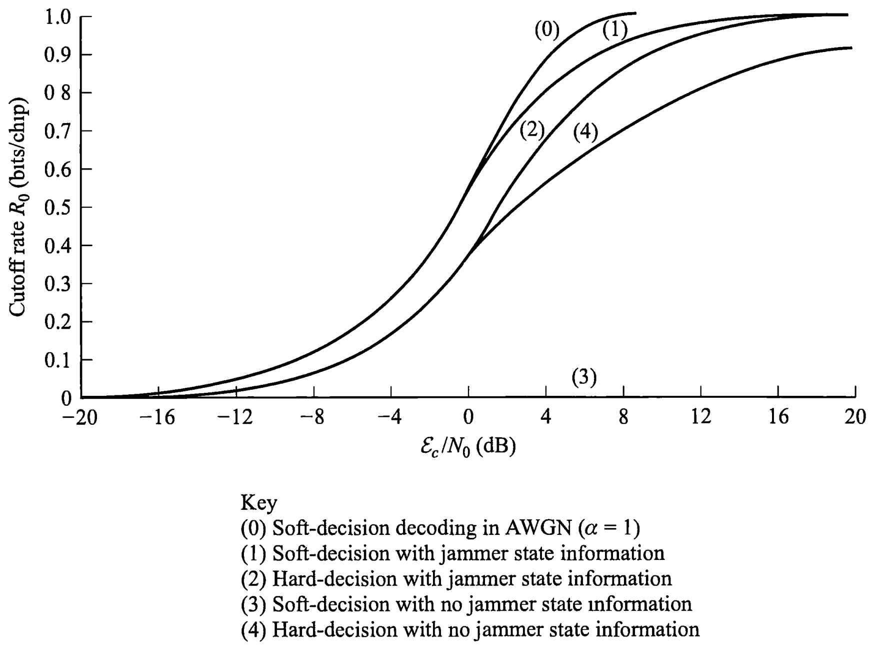


FIGURE 12.2–14

Cutoff rate for coded DS binary PSK modulation. [*From Omura and Levitt (1982).* © *1982 IEEE*].

Since the code is rate 1/2, we have

$$\frac{\mathcal{E}_c}{N_0} = 2\,\mathrm{dB}$$

Now, we go to the graphs in Figure 12.2–14 and find that for the AWGN channel (reference system) with $\mathcal{E}_c/N_0 = 2$ dB, the corresponding value of the cutoff rate is

$$R_0 = 0.74 \text{ bit per symbol}$$

If we have another channel with different noise characteristics (a worst-case pulse noise channel) but with the same value of the cutoff rate R_0, then the upper bound on the bit error probability is the same, i.e., 10^{-6} in this case. Consequently, we can use this rate to determine the SNR required for the worst-case pulse jammer channel. From the graphs in Figure 12.2–14, we find that

$$\frac{\mathcal{E}_c}{J_0} = \begin{cases} 10\ \mathrm{dB} & \text{for hard-decision decoding with} \\ & \text{no knowledge of jammer state} \\ 5\ \mathrm{dB} & \text{for hard-decision decoding with} \\ & \text{knowledge of jammer state} \\ 3\ \mathrm{dB} & \text{for soft-decision decoding with} \\ & \text{knowledge of jammer state} \end{cases}$$

Therefore, the corresponding values of $\mathcal{E}_b/J_0$ for the rate 1/2, $K = 7$ convolutional code are 13, 8, and 6 dB, respectively.

This general approach may be used to generate error rate graphs for coded binary signals in a worst-case pulse jamming channel by using corresponding error rate graphs for the AWGN channel. The approach we describe above is easily generalized to M-ary coded signals as indicated by Omura and Levitt (1982).

By comparing the cutoff rate for coded DS binary PSK modulation shown in Figure 12.2–14, we note that for rates below 0.7, there is no penalty in SNR with soft-decision decoding and jammer state information compared with the performance on the AWGN channel ($\alpha = 1$). On the other hand, at $R_0 = 0.7$, there is a 6-dB difference in performance between the SNR in an AWGN channel and that required for hard-decision decoding with no jammer state information. At rates below 0.4, there is no penalty in SNR with hard-decision decoding if the jammer state is unknown. However, there is the expected 2-dB loss in hard-decision decoding compared with soft-decision decoding in the AWGN channel.

12.2–4 Excision of Narrowband Interference in DS Spread Spectrum Systems

We have shown that DS spread spectrum signals reduce the effects of interference due to other users of the channel and intentional jamming. When the interference is narrowband, the cross correlation of the received signal with the replica of the PN code sequence reduces the level of the interference by spreading it across the frequency band occupied by the PN signal. Thus, the interference is rendered equivalent to a lower-level noise with a relatively flat spectrum. Simultaneously the cross correlation operation collapses the desired signal to the bandwidth occupied by the information signal prior to spreading. Consequently, the power in the narrowband interference is reduced by an amount equal to the processing gain.

The interference immunity of a DS spread spectrum communication system corrupted by narrowband interference can be further improved by filtering (whitening) the signal prior to despreading, where the objective is to reduce the level of the interference at the expense of introducing some distortion on the desired signal. This filtering can be accomplished by exploiting the wideband spectral characteristics of the desired DS signal and the narrowband characteristic of the interference as described below.

To be specific, we consider the demodulator illustrated in Figure 12.2–15. The received signal is passed through a filter matched to the chip pulse $g(t)$. The output of

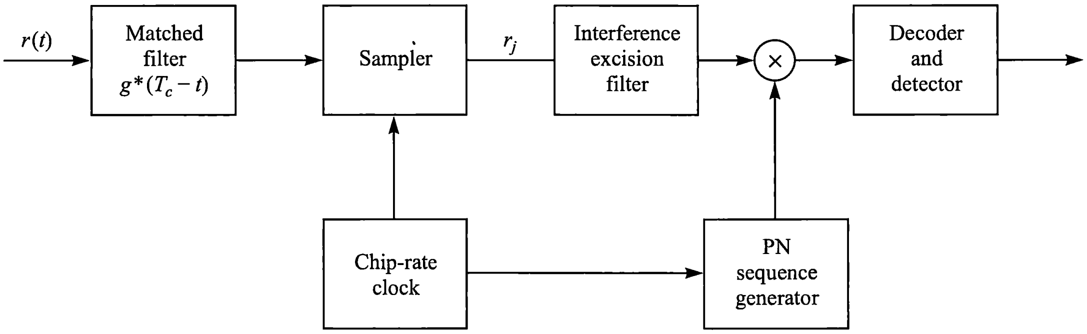


FIGURE 12.2–15
Demodulator for PN spread spectrum signal corrupted by narrowband interference.

this filter is synchronously sampled every T_c seconds to yield

$$r_j = 2\mathcal{E}_c(2b_j - 1)(2c_{ij} - 1) + \nu_j, \qquad j = 1, 2, \ldots \tag{12.2–60}$$

where $\mathcal{E}_c$ is the energy of the chip pulse, $\{b_j\}$ is the binary-valued PN sequence, and ν_j represents the additive noise and interference term. The additive noise term ν_j will be assumed to consist of two terms, one corresponding to a broadband noise (usually thermal noise) and the other to narrowband interference. Consequently we may express r_j as

$$r_j = s_j + i_j + n_j \tag{12.2–61}$$

where s_j denotes the signal component, i_j the narrowband interference, and n_j the broadband noise.

The received signal sequence $\{r_j\}$ at the output of the sampler is fed to a discrete-time filter that estimates the narrowband interference sequence $\{i_j\}$ and subtracts the estimate $\hat{i}_j$ from $\{r_j\}$. This filter may be either linear or non-linear. The resulting signal sequence $\{r_j - \hat{i}_j\}$ is then fed to the PN correlator, whose output is passed to the decoder.

Interference estimation and suppression based on linear prediction The interference component i_j can be estimated from the received signal by passing it through the linear transversal filter. Computationally efficient algorithms based on linear prediction may be used to estimate the interference. Basically, in this method the narrowband interference is modeled as having been generated by passing white noise through an all-pole filter. Hence, the output of this filter is an autoregressive (AR) process. Linear prediction is used to estimate the coefficients of the all-pole model. The estimated coefficients specify an appropriate noise-whitening all-zero (transversal) filter which is used to suppress the narrowband interference.

Let us assume for the moment that the statistics of the sequence $\{i_j\}$ are known and that $\{i_j\}$ is a stationary random sequence. Then, because of the narrowband characteristics of $\{i_j\}$, we can predict i_j from $r_{j-1}, r_{j-2}, \ldots, r_{j-m}$. That is,

$$\hat{i}_j = \sum_{l=1}^{m} a_{ml} r_{j-l} \tag{12.2–62}$$

where $\{a_{ml}\}$ are the coefficients of an mth-order linear predictor. It should be emphasized that Equation 12.2–62 predicts the interference but not the signal s_j, because the PN chips are uncorrelated and, hence, s_j is uncorrelated with r_{j-l}, $l = 1, 2, \ldots, m$, where m is less than the length of the PN sequence.

The coefficients in Equation 12.2–62 are determined by minimizing the mean square error between r_j and $\hat{i}_j$, with respect to the predictor coefficients. This leads to the set of linear equations, called the Yule–Walker equations,

$$\sum_{l=1}^{m} a_{ml} R(k - l) = R(k), \qquad k = 1, 2, \ldots, m \tag{12.2–63}$$

where $R(k) = E(r_j r_{j+k})$ is the autocorrelation function of the received signal $\{r_j\}$.

The solution of Equation 12.2–63 for the coefficients of the prediction filter requires knowledge of the autocorrelation function $R(k)$. In practice, the autocorrelation function of $\{i_j\}$ and, hence, $\{r_j\}$ is usually unknown, and it may also be slowly varying in time (nonstationary interference). In such a case, adaptive algorithms may be used to estimate the narrowband interference. In particular, least-squares-type algorithms, such as the Burg algorithm, are especially effective for estimating the coefficients of the linear prediction filter adaptively, as described in the paper by Ketchum and Proakis (1982).

EXAMPLE 12.2–4. Let us consider a narrowband interference that occupies 20 percent of the spectral band occupied by the PN spread spectrum signal. The average power of the interference is 20 dB above the average power of the signal. The average power of the broadband noise is 20 dB below the average power of the signal. Figure 12.2–16 illustrates the spectral characteristics of a 16-tap and a 29-tap FIR filter when the interference is equally split into four frequency bands. It is apparent that the 29-tap filter has better spectral characteristics. In general, the number of taps in the filter should be about four times the number of interference bands for adequate suppression. It is also apparent that the interference suppression filter acts as a notch filter. In effect, it attempts to whiten the total noise plus interference, so that the power spectral density of these components at its output is approximately flat. While suppressing the interference, the filter also distorts the desired signal by spreading it in time.

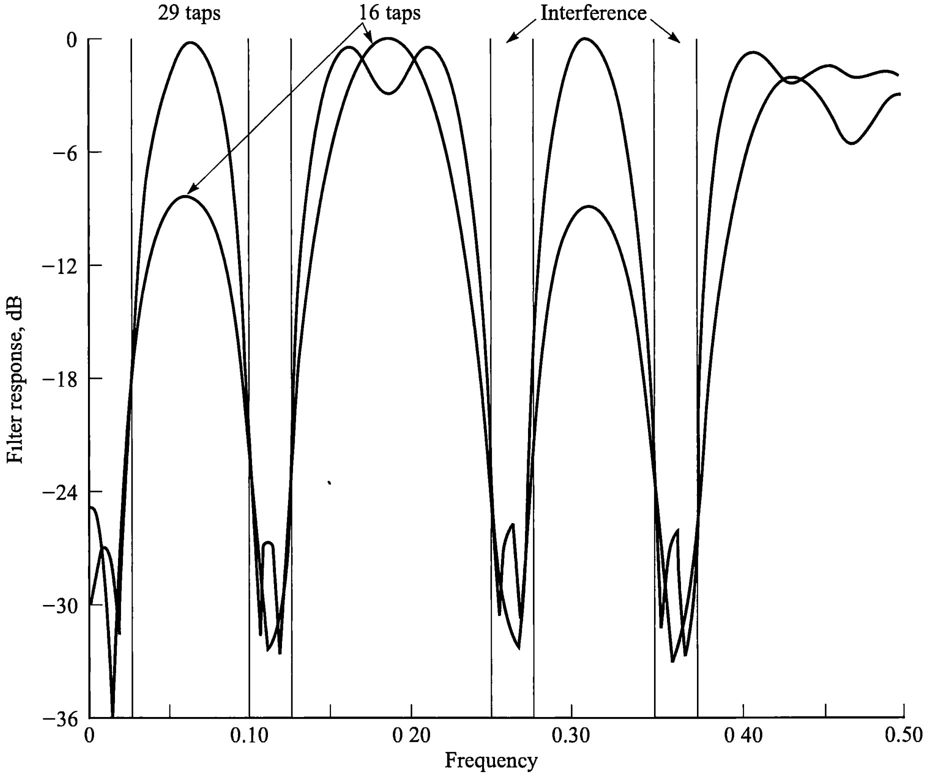


FIGURE 12.2–16
Frequency-response characteristics of 16- and 29-tap filters for four bands of interference.

Performance improvement with interference suppression Since the noise plus interference at the output of the suppression filter is spectrally flat, the matched filtering or cross correlation following the suppression filter should be performed with the distorted signal. This may be accomplished by having a filter matched to the interference suppression filter, i.e., a discrete-time filter impulse response $\{-a_{m,m}, -a_{m,m-1} \ldots - a_{m,1}, 1\}$ followed by the PN correlator. In fact, we can combine the interference suppression filter and its matched filter into a single filter having an impulse response

$$\begin{aligned}
h_0 &= -a_{m,m} \\
h_k &= -a_{m,m-k} + \sum_{l=0}^{k-1} a_{m,m-l} a_{m,k-l}, \qquad 1 \le k \le m-1 \\
h_m &= 1 + \sum_{l=1}^{m} a_{m,l}^2 \\
h_{m+k} &= h_{m-k}, \qquad 0 \le k \le m
\end{aligned} \tag{12.2–64}$$

The combined filter is a linear phase (symmetric) transversal filter with $K = 2m + 1$ taps. The impulse response may be normalized by dividing every term by h_m. Thus the center tap is normalized to unity. In order to demonstrate the effectiveness of the interference suppression filter, we compare the performance of the DS system with and without the suppression filter. The output SNR is a convenient performance index for this purpose. Since the output of the PN correlator is characterized as Gaussian, there is a one-to-one correspondence between the SNR and the probability of error.

Without the suppression filter, the PN correlator output , denoted as U_1, has mean $2\mathcal{E}_c L_c$ and a variance $L_c(2\mathcal{E}_c N_0 + R_{ii}(0))$ where $R_{ii}(k)$ is the autocorrelation function of the sequence $\{i_j\}$ and L_c is the number of chips per bit or per symbol. The output SNR is defined as the ratio of the square of the mean to twice the variance. Hence the SNR without the suppression filter is

$$\text{SNR}_{no} = \frac{\mathcal{E}_c L_c}{N_0 + R_{ii}(0)/2\mathcal{E}_c} \tag{12.2–65}$$

With an interference suppression filter having a symmetric impulse response as defined in Equation 12.2–64 and normalized such that the center tap is unity, the mean value of the correlator output is also $2\mathcal{E}_c L_c$. However, the variance of the output now consists of three terms. One corresponds to the additive wideband noise, the second to the residual narrowband interference, and the third to a self-noise caused by the time dispersion introduced by the suppression filter. The expression for the variance can be shown to be (see Ketchum and Proakis [1982]):

$$\begin{aligned}
\text{VAR}[U_1] = {} & 2L_c\mathcal{E}_c N_0 \sum_{k=0}^{K} h_k^2 + L_c \sum_{k=0}^{K}\sum_{l=0}^{K} h(l)h(k)R_{ii}(k-l) \\
& + 4L_c\mathcal{E}_c^2 \sum_{k=0}^{K/2-1} \left(2 - \frac{k}{L_c}\right) h_k^2
\end{aligned} \tag{12.2–66}$$

Hence the output SNR with the filter is the ratio of the square of the mean to twice the variance. The ratio of the SNR with the filter to the SNR without the filter is

$$\eta_o = \frac{N_0 + R_{ii}(0)/2\mathcal{E}_c}{N_0 \sum_{k=0}^{K} h_k^2 + \frac{1}{2\mathcal{E}_c} \sum_{k=0}^{K} \sum_{l=0}^{K} h(k)h(l)R_{ii}(k-l) + 2\mathcal{E}_c \sum_{k=0}^{K/2-1} (2 - k/L_c)h_k^2} \tag{12.2-67}$$

This ratio is called the improvement factor resulting from interference suppression. It may be plotted against the normalized SNR per chip without filtering, defined as

$$\frac{\mathrm{SNR}_{no}}{L_c} = \frac{\mathcal{E}_c}{N_0 + R_{ii}(0)/2\mathcal{E}_c} \tag{12.2-68}$$

The resulting graph of η_o versus SNR_{no}/L_c is universal in the sense that it applies to any PN spread spectrum system with arbitrary processing gain for a given $\mathcal{E}_c$, N_0, and $R_{ii}(0)$.

As an example, the improvement factor in (decibels) is plotted against SNR_{no}/L_c in Figure 12.2–17 for a single-band equal-amplitude randomly phased sinusoids covering 20 percent of the frequency band occupied by the DS spread spectrum signal. The interference suppression filter consists of a nine-tap suppression filter which corresponds to a fourth-order predictor. These numerical results indicate that the notch filter is very effective in suppressing the interference prior to PN correlation and decoding. As a consequence, the interference margin of the system is increased.

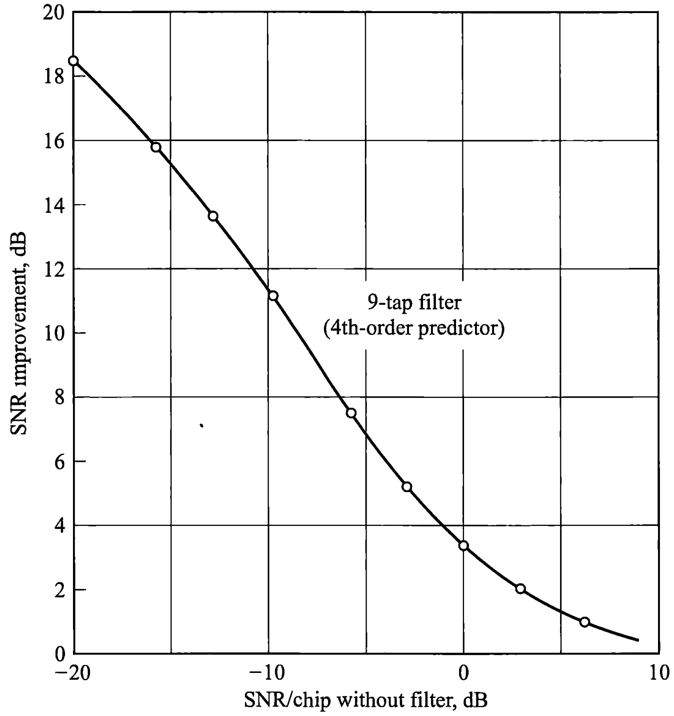


FIGURE 12.2–17
Improvement factor for interference suppression filter in cascade with its matched filter.

The use of a linear adaptive FIR filter for suppression of narrowband interference in DS spread spectrum systems has been considered in the literature by many authors. The interested reader is referred to this literature cited in Section 12.6. A practical motivation for excision of narrowband signals from wideband signals is to allow the overlay of narrowband digital cellular systems with wideband CDMA systems.

Interference estimation and suppression based on non-linear filtering The linear FIR filter used to predict the narrowband interference, which is modeled as a Gaussian autoregressive (AR) process, is the optimal minimum mean-square-error filter when the signal $\{s_k\}$ and broadband noise $\{n_k\}$ components are Gaussian random processes. However, the DS spread spectrum signal sequence $\{s_k\}$ is non-Gaussian. Consequently, the linear estimation filter is suboptimal, in the sense that it is not the best filter for suppressing the narrowband interference. The optimum estimator for the narrowband interference is non-linear.

By defining the state vector $\boldsymbol{x}_k$ as

$$\boldsymbol{x}_k = [i_k \quad i_{k-1} \quad \cdots \quad i_{k-m+1}]^t \tag{12.2–69}$$

where m is the order of the AR model, it is possible to express the state vector and the observation sequence in the state-space form

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{\Phi} \boldsymbol{x}_{k-1} + \boldsymbol{w}_k \\ r_k &= \boldsymbol{H} \boldsymbol{x}_k + (n_k + s_k) \end{aligned} \tag{12.2–70}$$

where $\boldsymbol{\Phi}$ is the state transition matrix that depends on the AR model parameters, $\boldsymbol{w}_k$ is the white Gaussian process driving the AR model, and $\boldsymbol{H} = [100\ldots0]$. We recall that the minimum mean-square-error estimator for the state at time k given the observations $\boldsymbol{r}_{k-1} \equiv [r_{k-1}, r_{k-2}, \ldots, r_0]$ is the conditional mean $E(\boldsymbol{x}_k|\boldsymbol{r}_{k-1})$. If the signal sequence $\{s_k\}$ and the broadband noise sequence $\{n_k\}$ were Gaussian, the optimum estimator for the state $\boldsymbol{x}_k$ corresponding to the conditional mean would be the linear predictor obtained from the Kalman filter. Since $\{s_k\}$ is non-Gaussian, the conditional mean estimate is a non-linear function of the observations which, in general, is highly complex. However, it is possible to derive a reduced complexity approximation to the conditional mean estimate. This approach has been described in the papers by Vijayan and Poor (1990), Garth and Poor (1992), Rusch and Poor (1994), and Poor and Rusch (1994). The general configuration of the approximate conditional mean non-linear filter is shown in Figure 12.2–18. The non-linear function $\tanh(x)$ provides a soft-decision type feedback signal component. An analysis and simulation results of the performance of this type of non-linear filter for suppression of narrowband interference are given in the papers cited above.

12.2–5 Generation of PN Sequences

The generation of PN sequences for spread spectrum applications is a topic that has received considerable attention in the technical literature. We shall briefly discuss the construction of some PN sequences and present a number of important properties of the

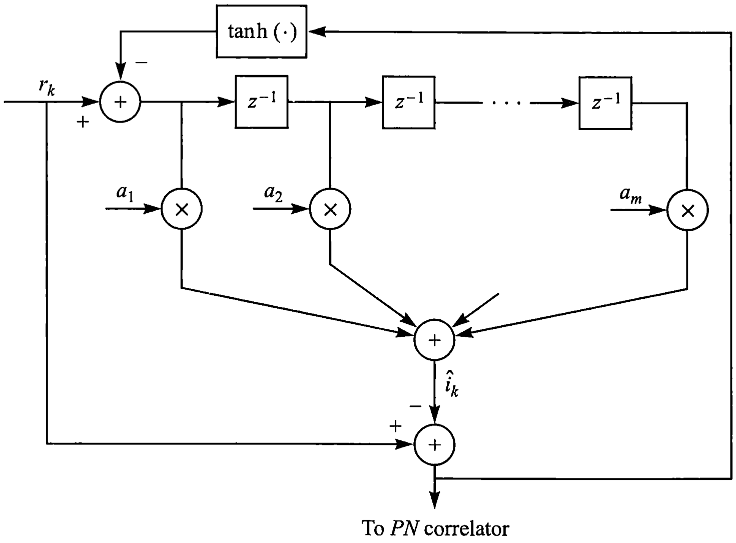


FIGURE 12.2–18
Non-linear excision filter.

autocorrelation and cross-correlation functions of such sequences. For a comprehensive treatment of this subject, the interested reader may refer to the book by Golomb (1967).

By far the most widely known binary PN sequences are the maximum-length shift-register sequences introduced in Section 7.9–5 in the context of coding. A maximum-length shift register sequence, or m-sequence for short, has length $n = 2^m - 1$ bits and is generated by an m-stage shift register with linear feedback as illustrated in Figure 12.2–19. The sequence is periodic with period n. Each period of the sequence contains 2^{m-1} ones and $2^{m-1} - 1$ zeros.

In DS spread spectrum applications the binary sequence with elements $\{0, 1\}$ is mapped into a corresponding sequence of positive and negative pulses according to the relation

$$p_i(t) = (2b_i - 1)p(t - iT)$$

where $p_i(t)$ is the pulse corresponding to the element b_i in the sequence with elements $\{0, 1\}$. Equivalently, we may say that the binary sequence with elements $\{0, 1\}$ is mapped into a corresponding binary sequence with elements $\{-1, 1\}$. We shall call the equivalent

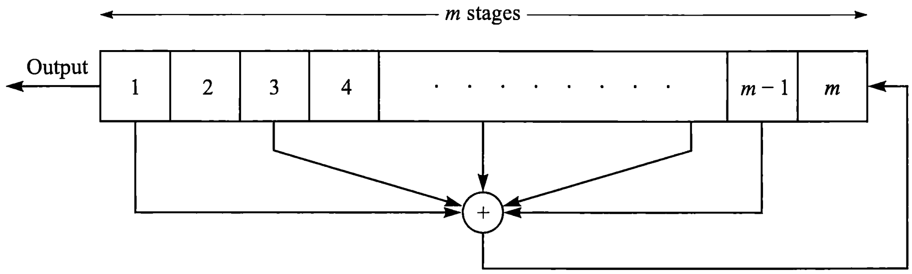


FIGURE 12.2–19
General m-stage shift register with linear feedback.

sequence with elements $\{-1, 1\}$ a *bipolar sequence*, since it results in pulses of positive and negative amplitudes.

An important characteristic of a periodic PN sequence is its periodic autocorrelation function, which is usually defined in terms of the bipolar sequence as

$$R(j) = \sum_{i=1}^{n} (2b_i - 1)(2b_{i+j} - 1), \qquad 0 \le j \le n-1 \tag{12.2–71}$$

where n is the period. Clearly, $R(j + rn) = R(j)$ for any integer value r.

Ideally, a pseudorandom sequence should have an autocorrelation function with the property that $R(0) = n$ and $R(j) = 0$ for $1 \le j \le n-1$. In the case of m sequences, the periodic autocorrelation function is

$$R(j) = \begin{cases} n & j = 0 \\ -1 & 1 \le j \le n-1 \end{cases} \tag{12.2–72}$$

For large values of n, i.e., for long m sequences, the size of the off-peak values of $R(j)$ relative to the peak value $R(j)/R(0) = -1/n$ is small and, from a practical viewpoint, inconsequential. Therefore, m sequences are almost ideal when viewed in terms of their autocorrelation function.

In antijamming applications of PN spread spectrum signals, the period of the sequence must be large in order to prevent the jammer from learning the feedback connections of the PN generator. However, this requirement is impractical in most cases because the jammer can determine the feedback connections by observing only $2m - 1$ chips from the PN sequence. This vulnerability of the PN sequence is due to the linearity property of the generator. To reduce the vulnerability to a jammer, the output sequences from several stages of the shift register or the outputs from several distinct m sequences are combined in a non-linear way to produce a non-linear sequence that is considerably more difficult for the jammer to learn. Further reduction in vulnerability is achieved by frequently changing the feedback connections and/or the number of stages in the shift register according to some prearranged plan formulated between the transmitter and the intended receiver.

In some applications, the cross-correlation properties of PN sequences are as important as the autocorrelation properties. For example, in CDMA, each user is assigned a particular PN sequence. Ideally, the PN sequences among users should be mutually orthogonal so that the level of interference experienced by any one user from transmissions of other users adds on a power basis. However, the PN sequences used in practice exhibit some correlation.

To be specific, we consider the class of m sequences. It is known (Sarwate and Pursley, 1980) that the periodic cross-correlation function between any pair of m sequences of the same period can have relatively large peaks. Table 12.2–1 lists the peak magnitude $R_{\max}$ for the periodic cross correlation between pairs of m sequences for $3 \le m \le 12$. The table also shows the number of m sequences of length $n = 2^m - 1$ for $3 \le m \le 12$. As we can see, the number of m sequences of length n increases rapidly with m. We also observe that, for most sequences, the peak magnitude $R_{\max}$ of the cross-correlation function is a large percentage of the peak value of the autocorrelation function.

TABLE 12.2–1
Peak Cross Correlation of m Sequences and Gold Sequences

m	$n = 2^m - 1$	Number of m sequences	Peak cross correlation $R_{\max}$	$R_{\max}/R(0)$	$t(m)$	$t(m)/R(0)$
3	7	2	5	0.71	5	0.71
4	15	2	9	0.60	9	0.60
5	31	6	11	0.35	9	0.29
6	63	6	23	0.36	17	0.27
7	127	18	41	0.32	17	0.13
8	255	16	95	0.37	33	0.13
9	511	48	113	0.22	33	0.06
10	1023	60	383	0.37	65	0.06
11	2047	176	287	0.14	65	0.03
12	4095	144	1407	0.34	129	0.03

Such high values for the cross correlations are undesirable in CDMA. Although it is possible to select a small subset of m sequences that have relatively smaller cross-correlation peak values, the number of sequences in the set is usually too small for CDMA applications.

PN sequences with better periodic cross-correlation properties than m sequences have been given by Gold (1967, 1968) and Kasami (1966). They are derived from m sequences as described below.

Gold and Kasami proved that certain pairs of m sequences of length n exhibit a three-valued cross-correlation function with values $\{-1, -t(m), t(m) - 2\}$, where

$$t(m) = \begin{cases} 2^{(m+1)/2} + 1 & \text{odd } m \\ 2^{(m+2)/2} + 1 & \text{even } m \end{cases} \tag{12.2–73}$$

For example, if $m = 10$, then $t(10) = 2^6 + 1 = 65$ and the three possible values of the periodic cross-correlation function are $\{-1, -65, 63\}$. Hence the maximum cross correlation for the pair of m sequences is 65, while the peak for the family of 60 possible sequences generated by a 10-stage shift register with different feedback connections is $R_{\max} = 383$—about a sixfold difference in peak values. Two m sequences of length n with a periodic cross-correlation function that takes on the possible values $\{-1, -t(m), t(m) - 2\}$ are called *preferred sequences*.

From a pair of preferred sequences, say $\boldsymbol{a} = [a_1\, a_2 \cdots a_n]$ and $\boldsymbol{b} = [b_1\, b_2 \cdots b_n]$, we construct a set of sequences of length n by taking the modulo-2 sum of $\boldsymbol{a}$ with the n cyclicly shifted versions of $\boldsymbol{b}$ or vice versa. Thus, we obtain n new periodic sequences† with period $n = 2^m - 1$. We may also include the original sequences $\boldsymbol{a}$ and $\boldsymbol{b}$, and, thus, we have a total of $n + 2$ sequences. The $n + 2$ sequences constructed in this manner are called *Gold sequences*.

---

†An equivalent method for generating the n new sequences is to employ a shift register of length $2m$ with feedback connections specified by the polynomial $h(X) = h_1(X)h_2(X)$, where $h_1(X)$ and $h_2(X)$ are the polynomials that specify the feedback connections of the m-stage shift registers that generate the m sequences $\boldsymbol{a}$ and $\boldsymbol{b}$.

EXAMPLE 12.2-5. Let us consider the generation of Gold sequences of length $n = 31 = 2^5 - 1$. As indicated above for $m = 5$, the cross-correlation peak is

$$t(5) = 2^3 + 1 = 9$$

Two preferred sequences, which may be obtained from Peterson and Weldon (1972), are described by the parity polynomials

$$h_1(X) = X^5 + X^3 + 1$$
$$h_2(X) = X^5 + X^4 + X^3 + X + 1$$

The shift registers for generating the two m sequences and the corresponding Gold sequences are shown in Figure 12.2–20. In this case, there are 33 different sequences, corresponding to the 33 relative phases of the two m sequences. Of these, 31 sequences are non-maximal-length sequences.

With the exception of the sequences $\boldsymbol{a}$ and $\boldsymbol{b}$, the set of Gold sequences is not comprised of maximum-length shift-register sequences of length n. Hence, their autocorrelation functions are not two-valued. Gold (1968) has shown that the cross-correlation function for any pair of sequences from the set of $n + 2$ Gold sequences is three-valued with possible values $\{-1, -t(m), t(m) - 2\}$, where $t(m)$ is given by Equation 12.2–73. Similarly, the off-peak autocorrelation function for a Gold sequence takes on values from the set $\{-1, -t(m), t(m) - 2\}$. Hence, the off-peak values of the autocorrelation function are upper-bounded by $t(m)$.

The values of the off-peak autocorrelation function and the peak cross-correlation function, i.e., $t(m)$, for Gold sequences is listed in Table 12.2–1. Also listed are the values normalized by $R(0)$.

The frequency of occurrence for each of the three possible values of the cross correlation for any pair of Gold sequences may also be of interest to the system designer. In Table 12.2–2, we give the frequency of occurrence of the three values for the case in which m is odd.

It is interesting to compare the peak cross-correlation value of Gold sequences with a known lower bound on the cross-correlation between any pair of binary sequences of period n in a set of M sequences. A lower bound derived by Welch (1974) for

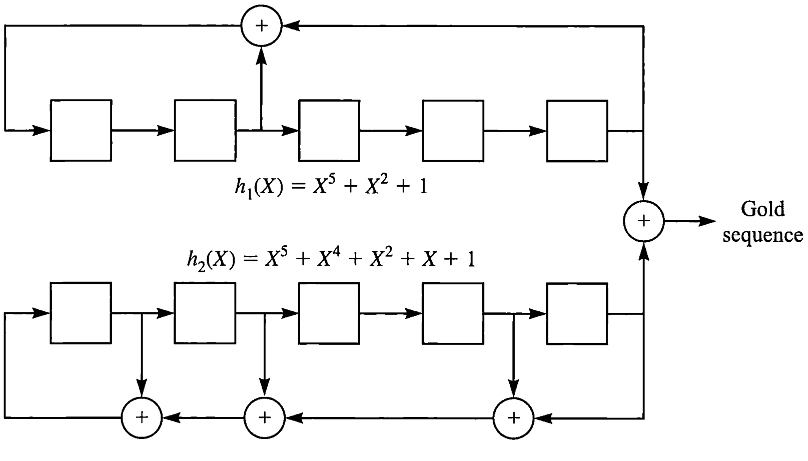


FIGURE 12.2–20
Generation of Gold sequences of length 31.

TABLE 12.2–2
Frequency of Occurrence of Cross-Correlation Values for Gold Codes of Length $n = 2^m - 1$, m Odd

Cross-correlation value	Frequency of occurrence
-1	$2^{n-1} - 1$
$-[2^{(m+1)/2} + 1]$	$2^{n-2} - 2^{(n-3)/2}$
$2^{(m+1)/2} - 1$	$2^{n-2} + 2^{(n-3)/2}$

$R_{\max}$ is

$$R_{\max} \geq n\sqrt{\frac{M-1}{Mn-1}} \tag{12.2–74}$$

which, for large values of n and M, is well approximated as $\sqrt{n}$. For Gold sequences, $M = 2^m + 1$, $n = 2^m - 1$ and the lower bound is $R_{\max} \approx 2^{m/2}$. This bound is lower by $\sqrt{2}$ for odd m and by 2 for even m relative to $R_{\max} = t(m)$ for Gold sequences. Therefore, Gold sequences do not achieve the lower bound.

A procedure similar to that used for generating Gold sequences will generate a smaller set of $M = 2^{m/2}$ binary sequences of period $n = 2^m - 1$, where m is even. In this procedure, we begin with an m sequence $\boldsymbol{a}$ and we form a binary sequence $\boldsymbol{b}$ by taking every $2^{m/2} + 1$ bit of $\boldsymbol{a}$. Thus, the sequence $\boldsymbol{b}$ is formed by decimating $\boldsymbol{a}$ by $2^{m/2} + 1$. It can be verified that the resulting sequence $\boldsymbol{b}$ is periodic with period $2^{m/2} - 1$. For example, if $m = 10$, the period of $\boldsymbol{a}$ is $n = 1023$ and the peroid of $\boldsymbol{b}$ is 31. Hence, if we observe 1023 bits of the sequence $\boldsymbol{b}$, we shall see 33 repetitions of the 31-bit sequence. Now, by taking $n = 2^m - 1$ bits of the sequences $\boldsymbol{a}$ and $\boldsymbol{b}$, we form a new set of sequences by adding, modulo-2, the bits from $\boldsymbol{a}$ and the bits from $\boldsymbol{b}$ and all $2^{m/2} - 2$ cyclic shifts of the bits from $\boldsymbol{b}$. By including $\boldsymbol{a}$ in the set, we obtain a set of $2^{m/2}$ binary sequences of length $n = 2^m - 1$. These are called *Kasami sequences*. The autocorrelation and cross-correlation functions of these sequences take on values from the set $\{-1, -(2^{m/2} + 1), 2^{m/2} - 1\}$. Hence, the maximum cross-correlation value for any pair of sequences from the set is

$$R_{\max} = 2^{m/2} + 1 \tag{12.2–75}$$

This value of $R_{\max}$ satisfies the Welch lower bound for a set of $2^{m/2}$ sequences of length $n = 2^m - 1$. Hence, the Kasami sequences are optimal.

Besides the well-known Gold and Kasami sequences, there are other binary sequences appropriate for CDMA applications. The interested reader may refer to the work of Scholtz (1979), Olsen (1977), and Sarwate and Pursley (1980).

Finally, we wish to indicate that, although we have discussed the periodic cross-correlation function between pairs of periodic sequences, many practical CDMA systems may use information bit durations that encompass only fractions of a periodic sequence. In such cases, it is the partial-period cross correlation between two sequences that is important. A number of papers deal with this problem, including those by Lindholm (1968), Wainberg and Wolf (1970), Fredricsson (1975), Bekir et al. (1978), and Pursley (1979).

12.3

FREQUENCY-HOPPED SPREAD SPECTRUM SIGNALS

In a *frequency-hopped* (FH) spread spectrum communication system the available channel bandwidth is subdivided into a large number of contiguous frequency slots. In any signaling interval, the transmitted signal occupies one or more of the available frequency slots. The selection of the frequency slot(s) in each signaling interval is made pseudorandomly according to the output from a PN generator. Figure 12.3–1 illustrates a particular FH pattern in the time-frequency plane.

A block diagram of the transmitter and receiver for an FH spread spectrum system is shown in Figure 12.3–2. The modulation is usually either binary or M-ary FSK. For example, if binary FSK is employed, the modulator selects one of two frequencies corresponding to the transmission of either a 1 or a 0. The resulting FSK signal is translated in frequency by an amount that is determined by the output sequence from the PN generator, which, in turn, is used to select a frequency that is synthesized by the frequency synthesizer. This frequency is mixed with the output of the modulator and the resultant frequency-translated signal is transmitted over the channel. For example, m bits from the PN generator may be used to specify $2^m - 1$ possible frequency translations.

At the receiver, we have an identical PN generator, synchronized with the receiver signal, which is used to control the output of the frequency synthesizer. Thus, the pseudorandom frequency translation introduced at the transmitter is removed at the receiver by mixing the synthesizer output with the received signal. The resultant signal is demodulated by means of an FSK demodulator. A signal for maintaining synchronism of the PN generator with the frequency-translated received signal is usually extracted from the received signal.

Although PSK modulation gives better performance than FSK in an AWGN channel, it is sometimes difficult to maintain phase coherence in the synthesis of the frequencies used in the hopping pattern and, also, in the propagation of the signal over the channel as the signal is hopped from one frequency to another over a wide bandwidth. Consequently, FSK modulation with noncoherent detection is often employed with FH spread spectrum signals.

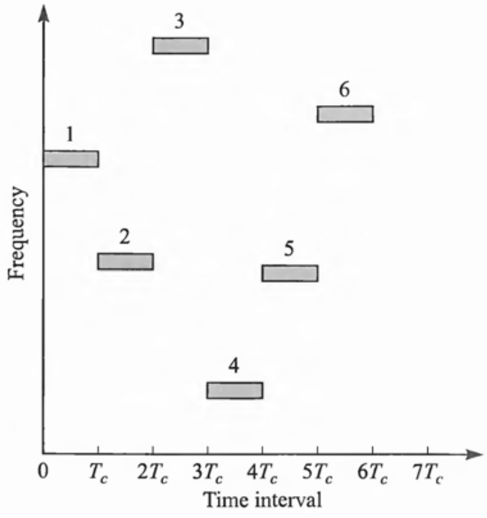


FIGURE 12.3–1
An example of a frequency-hopped (FH) pattern.

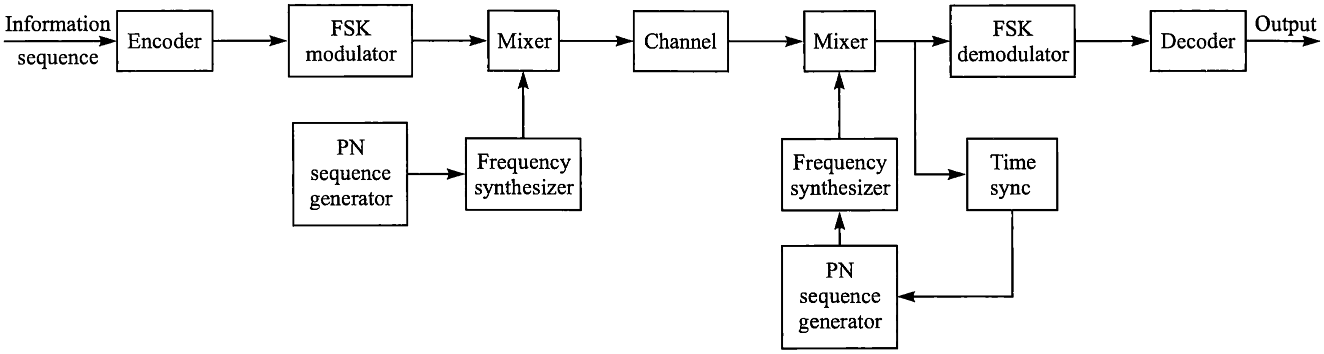

Information sequence
Encoder
FSK modulator
Mixer
Channel
Mixer
FSK demodulator
Decoder
Output
PN sequence generator
Frequency synthesizer
Frequency synthesizer
Time sync
PN sequence generator

FIGURE 12.3–2
Block diagram of an FH spread spectrum system.

In the FH system depicted in Figure 12.3–2, the carrier frequency is pseudorandomly hopped in every signaling interval. The M information-bearing tones are contiguous and separated in frequency by $1/T_c$, where T_c is the signaling interval. This type of frequency hopping is called *block hopping*.

Another type of frequency hopping that is less vulnerable to some jamming strategies is independent tone hopping. In this scheme, the M possible tones from the modulator are assigned widely dispersed frequency slots. One method for accomplishing this is illustrated in Figure 12.3–3. Here, the m bits from the PN generator and the k information bits are used to specify the frequency slots for the transmitted signal.

The FH rate is usually selected to be either equal to the (coded or uncoded) symbol rate or faster than that rate. If there are multiple hops per symbol, we have a fast-hopped signal. On the other hand, if the hopping is performed at the symbol rate, we have a slow-hopped signal.

Fast frequency hopping is employed in AJ applications when it is necessary to prevent a type of jammer, called a *follower jammer*, from having sufficient time to intercept the frequency and retransmit it along with adjacent frequencies so as to create interfering signal components. However, there is a penalty incurred in subdividing a signal into several FH elements because the energy from these separate elements is

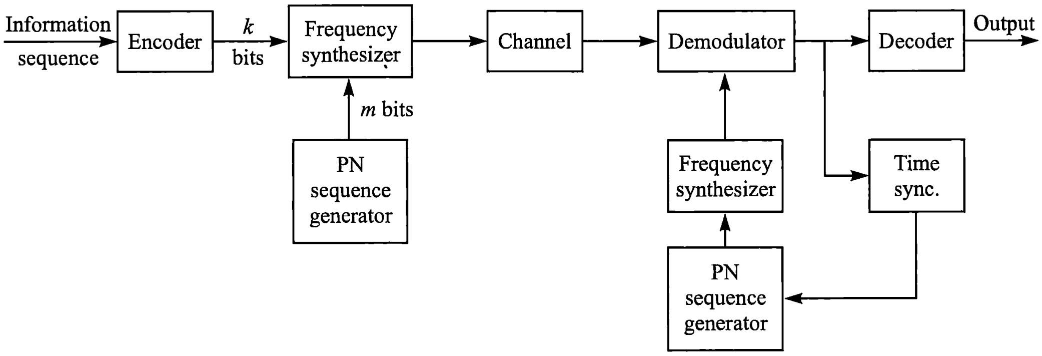

Information sequence
Encoder
k bits
Frequency synthesizer
m bits
PN sequence generator
Channel
Demodulator
Decoder
Output
Frequency synthesizer
Time sync.
PN sequence generator

FIGURE 12.3–3
Block diagram of an independent tone FH spread spectrum system.

combined noncoherently. Consequently, the demodulator incurs a penalty in the form of a noncoherent combining loss as described in Section 11.1.

FH spread spectrum signals are used primarily in digital communication systems that require AJ protection and in CDMA, where many users share a common bandwidth. In most cases, an FH signal is preferred over a DS spread spectrum signal because of the stringent synchronization requirements inherent in DS spread spectrum signals. Specifically, in a DS system, timing and synchronization must be established to within a fraction of the chip interval $T_c \approx 1/W$. On the other hand, in an FH system, the chip interval is the time spent in transmitting a signal in a particular frequency slot of bandwidth $B \ll W$. But this interval is approximately $1/B$, which is much larger than $1/W$. Hence the timing requirements in an FH system are not as stringent as in a DS system.

In Sections 12.3–2 and 12.3–3, we shall focus on the AJ and CDMA applications of FH spread spectrum signals. First, we shall determine the error rate performance of an uncoded and a coded FH signal in the presence of broadband AWGN inteference. Then we shall consider a more serious type of interference that arises in AJ and CDMA applications, called *partial-band interference*. The benefits obtained from coding for this type of interference are determined. We conclude the discussion in Section 12.3–3 with an example of an FH CDMA system that was designed for use by mobile users with a satellite serving as the channel.

12.3–1 Performance of FH Spread Spectrum Signals in an AWGN Channel

Let us consider the performance of an FH spread spectrum signal in the presence of broadband interference characterized statistically as AWGN with power spectral density J_0. For binary orthogonal FSK with noncoherent detection and slow frequency hopping (1 hop/bit), the probability of error, derived in Section 4.5–3, is

$$P_2 = \tfrac{1}{2}e^{-\gamma_b/2} \tag{12.3–1}$$

where $\gamma_b = \mathcal{E}_b/J_0$. On the other hand, if the bit interval is subdivided into L subintervals and FH binary FSK is transmitted in each subinterval, we have a fast FH signal. With square-law combining of the output signals from the corresponding matched filters for the L subintervals, the error rate performance of the FH signal, obtained from the results in Section 11.1, is

$$P_2(L) = \frac{1}{2^{2L-1}}e^{-\gamma_b/2}\sum_{i=0}^{L-1} K_i \left(\tfrac{1}{2}\gamma_b\right)^i \tag{12.3–2}$$

where the SNR per bit is $\gamma_b = \mathcal{E}_b/J_0 = L\gamma_c$, γ_c is the SNR per chip in the L-chip symbol, and

$$K_i = \frac{1}{i!}\sum_{r=0}^{L-1-i} \binom{2L-1}{r} \tag{12.3–3}$$

We recall that, for a given SNR per bit γ_b, the error rate obtained from Equation 12.3–2 is larger than that obtained from Equation 12.3–1. The difference in SNR for a given error rate and a given L is called the *noncoherent combining loss*, which was described and illustrated in Section 11.1.

Coding improves the performance of the FH spread spectrum system by an amount, which we call the *coding gain*, that depends on the code parameters. Suppose we use a linear binary (n, k) block code and binary FSK modulation with one hop per coded bit for transmitting the bits. With soft-decision decoding of the square-law-demodulated FSK signal, the probability of a codeword error is upper-bounded as

$$P_e \leq \sum_{m=2}^{M} P_2(m) \tag{12.3–4}$$

where $P_2(m)$ is the error probability in deciding between the mth codeword and the all-zero codeword when the latter has been transmitted. The expression for $P_2(m)$ was derived in Section 7.4 and has the same form as Equations 12.3–2 and 12.3–3, with L being replaced by w_m and γ_b by $\gamma_b R_c w_m$, where w_m is the weight of the mth code word and R_c is the code rate. The product $R_c w_m$, which is not less than $R_c d_{\min}$, represents the coding gain. Thus, we have the performance of a block coded FH system with slow frequency hopping in broadband interference.

The probability of error for fast frequency hopping with n_2 hops per coded bit is obtained by reinterpreting the binary event probability $P_2(m)$ in Equation 12.3–4. The n_2 hops per coded bit may be interpreted as a repetition code, which, when combined with a nontrivial (n_1, k) binary linear code having weight distribution $\{w_m\}$, yields an $(n_1 n_2, k)$ binary linear code having weight distribution $\{n_2 w_m\}$. Hence, $P_2(m)$ has the form given in Equation 12.3–2, with L replaced by $n_2 w_m$ and γ_b by $\gamma_b R_c n_2 w_m$, where $R_c = k/n_1 n_2$. Note that $\gamma_b R_c n_2 w_m = \gamma_b w_m k/n_1$, which is just the coding gain obtained from the nontrivial (n_1, k) code. Consequently, the use of the repetition code will result in an increase in the noncoherent combining loss.

With hard-decision decoding and slow frequency hopping, the probability of a coded bit error at the output of the demodulator for noncoherent detection is

$$p = \tfrac{1}{2} e^{-\gamma_b R_c/2} \tag{12.3–5}$$

The codeword error probability is easily upper bounded, by use of the Chernov bound, as

$$P_e \leq \sum_{m=2}^{M} [4p(1-p)]^{w_m/2} \tag{12.3–6}$$

However, if fast frequency hopping is employed with n_2 hops per coded bit, and the square-law-detected outputs from the corresponding matched filters for the n_2 hops are added as in soft-decision decoding to form the two decision variables for the coded bits, the bit error probability p is also given by Equation 12.3–2, with L replaced by n_2 and γ_b replaced by $\gamma_b R_c n_2$, where R_c is the rate of the nontrivial (n_1, k) code. Consequently, the performance of the fast FH system in broadband interference is degraded relative to the slow FH system by an amount equal to the noncoherent combining loss of the signals received from the n_2 hops.

We have observed that for both hard-decision and soft-decision decoding, the use of the repetition code in a fast FH system yields no coding gain. The only coding gain obtained comes from the (n_1, k) block code. Hence, the repetition code is inefficient in a fast FH system with noncoherent combining. A more efficient coding method is one in which either a single low-rate binary code or a concatenated code is employed. Additional improvements in performance may be obtained by using nonbinary codes in conjunction with M-ary FSK. Bounds on the error probability for this case may be obtained from the results given in Section 11.1.

Although we have evaluated the performance of linear block codes only in the above discussion, it is relatively easy to derive corresponding performance results for binary convolutional codes. We leave as an exercise for the reader the derivation of the bit error probability for soft-decision Viterbi decoding and hard-decision Viterbi decoding of FH signals corrupted by broadband interference.

Finally, we observe that $\mathcal{E}_b$, the energy per bit, can be expressed as $\mathcal{E}_b = P_{\text{av}}/R$, where R is the information rate in bits per second and $J_0 = J_{\text{av}}/2W$. Therefore, γ_b may be expressed as

$$\gamma_b = \frac{\mathcal{E}_b}{J_0} = \frac{2W/R}{J_{\text{av}}/P_{\text{av}}} \tag{12.3–7}$$

In this expression, we recognize W/R as the processing gain and $J_{\text{av}}/P_{\text{av}}$ as the interference margin for the FH spread spectrum signal.

12.3–2 Performance of FH Spread Spectrum Signals in Partial-Band Interference

The partial-band interference considered in this subsection is modeled as a zero-mean Gaussian random process with a flat power spectral density over a fraction α of the total bandwidth W and zero elsewhere. In the region or regions where the power spectral density is nonzero, its value is $R_{zz}(f) = 2J_0/\alpha, 0 < \alpha \leq 1$. This model of the interference may be applied to a jamming signal or to interference from other users in an FH CDMA system.

Suppose that the partial-band interference comes from a jammer who may select α to optimize the effect on the communication system. In an uncoded pseudorandomly hopped (slow-hopping) FH system with binary FSK modulation and noncoherent detection, the received signal will be jammed with probability α and it will not be jammed with probability $1-\alpha$. When it is jammed, the probability of error is $\frac{1}{2}$ exp $(-\mathcal{E}_b\alpha/2J_0)$, and when it is not jammed, the demodulation is error-free. Consequently, the average probability of error is

$$P_2(\alpha) = \tfrac{1}{2}\alpha \exp\left(-\frac{\alpha\mathcal{E}_b}{2J_0}\right) \tag{12.3–8}$$

where $\mathcal{E}_b/J_0$ may also be expressed as $(2W/R)/(J_{\text{av}}/P_{\text{av}})$.

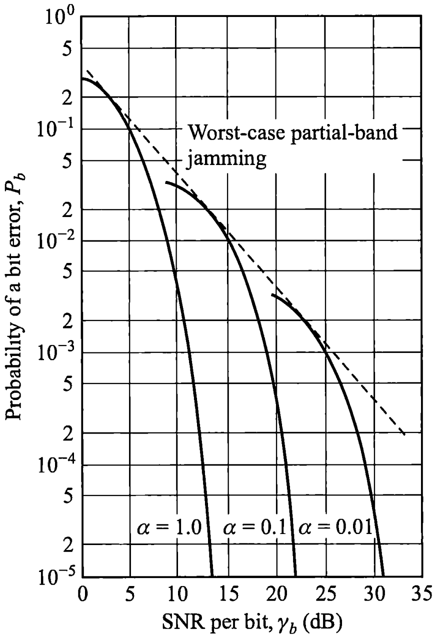


FIGURE 12.3–4
Performance of binary FSK with partial-band interference.

Figure 12.3–4 illustrates the error rate as a function of $\mathcal{E}_b/J_0$ for several values of α. The jammer's optimum strategy is to select the value of α that maximizes the error probability. By differentiating $P_2(\alpha)$ and solving for the extremum with the restriction that $0 \le \alpha \le 1$, we find that

$$\alpha^* = \begin{cases} \dfrac{1}{\mathcal{E}_b/2J_0} & \mathcal{E}_b/J_0 \ge 2 \\ 1 & \mathcal{E}_b/J_0 < 2 \end{cases} \tag{12.3–9}$$

The corresponding error probability for the worst-case partial-band jammer is

$$P_2 = \frac{e^{-1}}{\mathcal{E}_b/J_0} \tag{12.3–10}$$

Whereas the error probability decreases exponentially for full-band jamming, we now find that the error probability decreases only inversely with $\mathcal{E}_b/J_0$ for the worst-case partial-band jamming. This result is similar to the error rate performance of binary FSK in a Rayleigh fading channel (see Section 13.3) and to the uncoded DS spread spectrum system corrupted by worst-case pulse interference (see Section 12.2–3).

As we shall demonstrate below, signal diversity obtained by means of coding provides a significant improvement in performance relative to uncoded signals. This same approach to signal design is also effective for signaling over a fading channel, as we shall demonstrate in Chapter 13.

To illustrate the benefits of diversity in an FH spread spectrum signal with partial-band interference, we assume that the same information symbol is transmitted by binary FSK on L independent frequency hops. This may be accomplished by subdividing the signaling interval into L subintervals, as described previously for fast frequency hopping. After the hopping pattern is removed, the signal is demodulated by passing it through a pair of matched filters whose outputs are square-law-detected and sampled at the end of each subinterval. The square-law-detected signals corresponding to the L

frequency hops are weighted and summed to form the two decision variables (metrics), which are denoted as U_1 and U_2.

When the decision variable U_1 contains the signal components, U_1 and U_2 may be expressed as

$$\begin{aligned} U_1 &= \sum_{k=1}^{L} \beta_k |2\mathcal{E}_c + N_{1k}|^2 \\ U_2 &= \sum_{k=1}^{L} \beta_k |N_{2k}|^2 \end{aligned} \tag{12.3–11}$$

where $\{\beta_k\}$ represent the weighting coefficients, $\mathcal{E}_c$ is the signal energy per chip in the L-chip symbol, and $\{N_{jk}\}$ represent the additive Gaussian noise terms at the output of the matched filters.

The coefficients are optimally selected to prevent the interference from saturating the combiner should the transmitted frequencies be successfully hit in one or more hops. Ideally, β_k is selected to be equal to the reciprocal of the variance of the corresponding noise terms $\{N_k\}$. Thus, the noise variance for each chip is normalized to unity by this weighting and the corresponding signal is also scaled accordingly. This means that when the signal frequencies on a particular hop are interfered, the corresponding weight is very small. In the absence of interference on a given hop, the weight is relatively large. In practice, for partial-band interference, the weighting may be accomplished by use of an AGC having a gain that is set on the basis of noise power measurements obtained from frequency bands adjacent to the transmitted tones. This is equivalent to having side information (knowledge of jammer state) at the decoder.

Suppose that we have broadband Gaussian noise with power spectral density N_0 and partial-band interference, over αW of the frequency band, which is also Gaussian with power spectral density J_0/α. In the presence of partial-band interference, the variance of the real and imaginary parts of the noise terms N_{1k} and N_{2k} are

$$\sigma_k^2 = \tfrac{1}{2} E\left(|N_{1k}|^2\right) = \tfrac{1}{2} E\left(|N_{2k}|^2\right) = 2\mathcal{E}_c\left(N_0 + \frac{J_0}{\alpha}\right) \tag{12.3–12}$$

In this case, we select $\beta_k = 1/\sigma_k^2 = [2\mathcal{E}_c(N_0 + J_0/\alpha)]^{-1}$. In the absence of partial-band interference, $\sigma_k^2 = 2\mathcal{E}_c N_0$ and, hence, $\beta_k = (2\mathcal{E}_c N_0)^{-1}$. Note that β_k is a random variable. It is convenient to normalize the variance of the noise components to unity by defining, $N'_{1k} = \sqrt{\beta_k} N_{1k}$ and $N'_{2k} = \sqrt{\beta_k} N_{2k}$, where $\beta_k = 1/\sigma_k^2$ for the corresponding values of σ_k^2.

An error occurs in the demodulation if $U_2 > U_1$. Although it is possible to determine the exact error probability, we shall resort to the Chernov bound, which yields a result that is much easier to evaluate and interpret. Specifically, the Chernov (upper) bound on the error probability is

$$\begin{aligned} P_2 &= P(U_2 - U_1 > 0) \le E\{\exp[\nu(U_2 - U_1)]\} \\ &= E\left\{\exp\left[-\nu \sum_{k=1}^{L} \left(|2\sqrt{\beta_k}\mathcal{E}_c + N'_{1k}|^2 - |N'_{2k}|^2\right)\right]\right\} \end{aligned} \tag{12.3–13}$$

where $\nu > 0$ is a variable that is optimized to yield the tightest possible bound.

The averaging in Equation 12.3–13 is performed with respect to the statistics of the noise components and the statistics of the weighting coefficients $\{\beta_k\}$, which are random as a consequence of the statistical nature of the interference. Keeping the $\{\beta_k\}$ fixed and averaging over the noise statistics first, we obtain

$$\begin{aligned}
P_2(\boldsymbol{\beta}) &\le E\left[\exp\left(-\nu\sum_{k=1}^{L}|2\sqrt{\beta_k}\mathcal{E}_c + N'_{1k}|^2 + \nu\sum_{k=1}^{L}|N'_{2k}|^2\right)\right] \\
&= \prod_{k=1}^{L} E\left[\exp\left(-\nu|2\sqrt{\beta_k}\mathcal{E}_c + N'_{1k}|^2\right)\right]E\left[\exp\left(\nu|N'_{2k}|^2\right)\right] \\
&= \prod_{k=1}^{L}\frac{1}{1-4\nu^2}\exp\left(\frac{-4\mathcal{E}_c^2\beta_k\nu}{1+2\nu}\right) \qquad (12.3\text{–}14)
\end{aligned}$$

Since the FSK tones are interfered with probability α, it follows that $\beta_k = [2\mathcal{E}(N_0 + J_0/\alpha)]^{-1}$ with probability α and $(2\mathcal{E}_c N_0)^{-1}$ with probability $1-\alpha$. Hence, the Chernov bound is

$$\begin{aligned}
P_2 &\le \prod_{k=1}^{L}\left\{\frac{\alpha}{1-4\nu^2}\exp\left[\frac{-2\mathcal{E}_c\nu}{(N_0+J_0/\alpha)(1+2\nu)}\right] + \frac{1-\alpha}{1-4\nu^2}\exp\left[\frac{-2\mathcal{E}_c\nu}{N_0(1+2\nu)}\right]\right\} \\
&= \left\{\frac{\alpha}{1-4\nu^2}\exp\left[\frac{-2\mathcal{E}_c\nu}{(N_0+J_0/\alpha)(1+2\nu)}\right. + \frac{1-\alpha}{1-4\nu^2}\exp\left[\frac{-2\mathcal{E}_c\nu}{N_0(1+2\nu)}\right]\right\}^L \\
&\qquad (12.3\text{–}15)
\end{aligned}$$

The next step is to optimize the bound in Equation 12.3–15 with respect to the variable ν. In its present form, however, the bound is messy to manipulate. A significant simplification occurs if we assume that $J_0/\alpha, \ge N_0$, which renders the second term in Equation 12.3–15 negligible compared with the first. Alternatively, we let $N_0 = 0$, so that the bound on P_2 reduces to

$$P_2 \le \left\{\frac{\alpha}{1-4\nu^2}\exp\left[\frac{-2\alpha\nu\mathcal{E}_c}{J_0(1+2\nu)}\right]\right\}^L \qquad (12.3\text{–}16)$$

The minimum value of this bound with respect to ν and the maximum with respect to α (worst-case partial-band interference) is easily shown to occur when $\alpha = 3J_0/\mathcal{E}_c \le 1$ and $\nu = \frac{1}{4}$. For these values of the parameters, Equation 12.3–16 reduces to

$$P_2 \le P_2(L) = \left(\frac{4}{e\gamma_c}\right)^L = \left(\frac{1.47}{\gamma_c}\right)^L, \qquad \gamma_c = \frac{\mathcal{E}_c}{J_0} = \frac{\mathcal{E}_b}{LJ_0} \ge 3 \qquad (12.3\text{–}17)$$

where γ_c is the SNR per chip in the L-chip symbol.

The result in Equation 12.3–17 was first derived by Viterbi and Jacobs (1975).

We observe that the probability of error for the worst-case partial-band interference decreases exponentially with an increase in the SNR per chip γ_c. This result is very similar to the performance characteristics of diversity techniques for Rayleigh fading

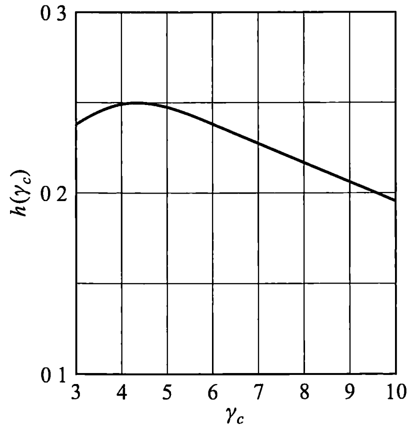


FIGURE 12.3–5
Graph of the function $h(\gamma_c)$.

channels (see Section 13.3). We may express the right-hand side of Equation 12.3–17 in the form

$$P_2(L) = \exp[-\gamma_b h(\gamma_c)] \tag{12.3–18}$$

where the function $h(\gamma_c)$ is defined as

$$h(\gamma_c) = -\frac{1}{\gamma_c}\left[\ln\left(\frac{4}{\gamma_c}\right) - 1\right] \tag{12.3–19}$$

A plot of $h(\gamma_c)$ is given in Figure 12.3–5. We observe that the function has a maximum value of $\frac{1}{4}$ at $\gamma_c = 4$. Consequently, there is an optimum SNR per chip of $10 \log \gamma_c =$ 6 dB. At the optimum SNR, the error rate is upper-bounded as

$$P_2 \le P_2(L_{\text{opt}}) = e^{-\gamma_b/4} \tag{12.3–20}$$

When we compare the error probability bound in Equation 12.3–20 with the error probability for binary FSK in spectrally flat noise, which is given by Equation 12.3–1, we see that the combined effect of worst-case partial-band interference and the noncoherent combining loss in the square-law combining of the L chips is 3 dB. We emphasize, however, that for a given $\mathcal{E}_b/J_0$, the loss is greater when the order of diversity is not optimally selected.

Coding provides a means for improving the performance of the FH system corrupted by partial-band interference. In particular, if a block orthogonal code is used, with $M = 2^k$ codewords and Lth-order diversity per codeword, the probability of a codeword error is upper-bounded as

$$P_e \le (2^k - 1)P_2(L) = (2^k - 1)\left(\frac{1.47}{\gamma_c}\right)^L = (2^k - 1)\left(\frac{1.47}{k\gamma_b/L}\right)^L \tag{12.3–21}$$

and the equivalent bit error probability is upper-bounded as

$$P_b \le 2^{k-1}\left(\frac{1.47}{k\gamma_b/L}\right)^L \tag{12.3–22}$$

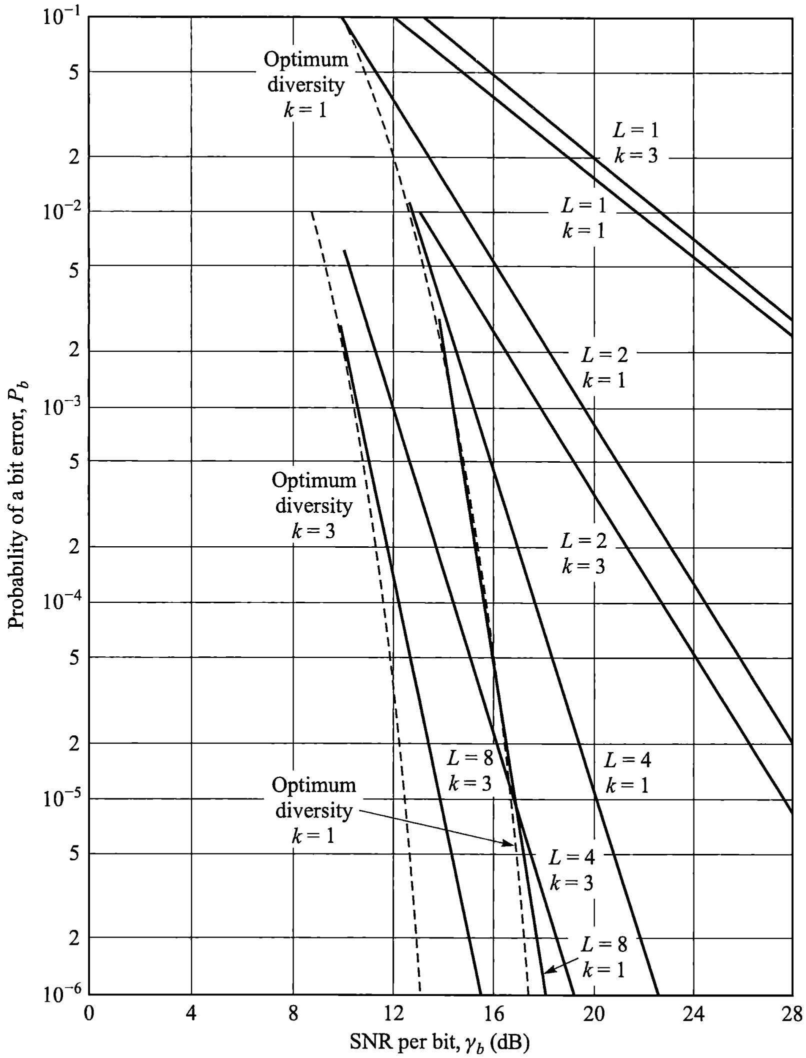


FIGURE 12.3–6
Performance of binary and octal FSK with L-order diversity for a channel with worst-case partial-band interference.

Figure 12.3–6 illustrates the probability of a bit error for $L = 1, 2, 4, 8$ and $k = 1, 3$. With an optimum choice of diversity, the upper bound can be expressed as

$$P_b \le 2^{k-1} \exp(-\tfrac{1}{4}k\gamma_b) = \tfrac{1}{2}\exp[-k(\tfrac{1}{4}\gamma_b - \ln 2)] \qquad (12.3\text{–}23)$$

Thus, we have an improvement in performance by an amount equal to $10\log[k(1 - 2.77/\gamma_b)]$. For example, if $\gamma_b = 10$ and $k = 3$ (octal modulation), then the gain is 3.4 dB, while if $k = 5$, then the gain is 5.6 dB.

Additional gains can be achieved by employing concatenated codes in conjunction with soft-decision decoding. In the example below, we employ a dual-k convolutional code as the outer code and a Hadamard code as the inner code on the channel with partial-band interference.

EXAMPLE 12.3-1. Suppose we use a Hadamard $H(n, k)$ constant weight code with on–off keying (OOK) modulation for each code bit. The minimum distance of the code is $d_{\min} = \frac{1}{2}n$, and, hence, the effective order of diversity obtained with OOK modulation is $\frac{1}{2}d_{\min} = \frac{1}{4}n$. There are $\frac{1}{2}n$ FH tones transmitted per code word. Hence,

$$\gamma_c = \frac{k}{\frac{1}{2}n}\gamma_b = 2R_c\gamma_b \tag{12.3–24}$$

when this code is used alone. The bit error rate performance for soft-decision decoding of these codes for the partial-band interference channel is upper-bounded as

$$P_b \le 2^{k-1}P_2(\tfrac{1}{2}d_{\min}) = 2^{k-1}\left(\frac{1.47}{2R_c\gamma_b}\right)^{n/4} \tag{12.3–25}$$

Now, if a Hadamard (n, k) code is used as the inner code and a rate 1/2 dual-k convolutional code (see Section 8.7) is the outer code, the bit error performance in the presence of worst-case partial-band interference is (see Equation 8.7–5)

$$P_b \le \frac{2^{k-1}}{2^k - 1}\sum_{m=4}^{\infty}\beta_m P_2(\tfrac{1}{2}md_{\min}) = \frac{2^{k-1}}{2^k - 1}\sum_{m=4}^{\infty}\beta_m P_2(\tfrac{1}{2}mn) \tag{12.3–26}$$

where $P_2(L)$ is given by Equation 12.3–17 with

$$\gamma_c = \frac{k}{n}\gamma_b = R_c\gamma_b \tag{12.3–27}$$

Figure 12.3–7 illustrates the performance of the dual-k codes for $k = 5, 4$, and 3 concatenated with the Hadamard $H(20, 5)$, $H(16, 4)$, and $H(12, 3)$ codes, respectively.

In the above discussion, we have focused on soft-decision decoding. On the other hand, the performance achieved with hard-decision decoding is significantly (several decibels) poorer than that obtained with soft-decision decoding. In a concatenated

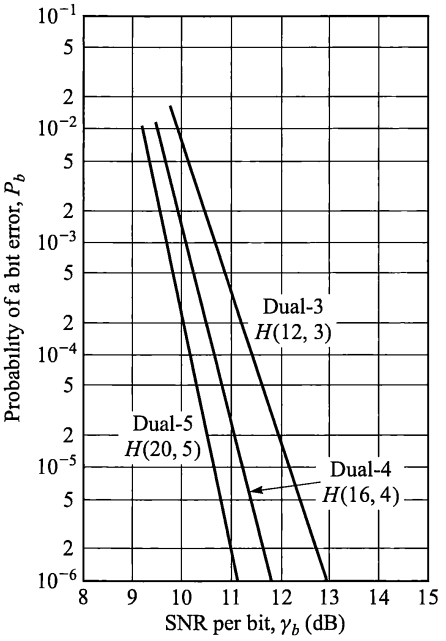


FIGURE 12.3–7
Performance of dual-k codes concatenated with Hadamard codes for a channel with worst-case partial-band interference.

coding scheme, however, a mixture involving soft decision decoding of the inner code and hard decision decoding of the outer code represents a reasonable compromise between decoding complexity and performance.

Finally, we wish to indicate that another serious threat in an FH spread spectrum system is partial-band multitone interference. This type of interference is similar in effect to partial-band spectrally flat noise interference. Diversity obtained through coding is an effective means for improving the performance of the FH system. An additional improvement is achieved by properly weighting the demodulator outputs so as to suppress the effects of the interference.

12.3–3 A CDMA System Based on FH Spread Spectrum Signals

In Section 12.2–2, we considered a CDMA system based on the use of DS spread spectrum signals. As previously indicated, it is also possible to have a CDMA system based on FH spread spectrum signals. Each transmitter–receiver pair in such a system is assigned its own pseudorandom FH pattern. Aside from this distinguishing feature, the transmitters and receivers of all the users may be identical in that they may have identical encoders, decoders, modulators, and demodulators.

CDMA systems based on FH spread spectrum signals are particularly attractive for mobile (land, air, sea) users because timing requirements are not as stringent as in a DS spread spectrum signal. In addition, frequency synthesis techniques and associated hardware have been developed that make it possible to frequency-hop over bandwidths that are significantly larger than those currently possible with DS spread spectrum systems. Consequently, larger processing gains are possible with FH. The capacity of CDMA with FH is also relatively high. Viterbi (1978) has shown that with dual-k codes and M-ary FSK modulation, it is possible to accommodate up to $\frac{3}{8}W/R$ simultaneous users who transmit at an information rate R bits/s over a channel with bandwidth W.

One of the earliest CDMA systems based on FH coded spread spectrum signals was built to provide multiple-access tactical satellite communications for small mobile (land, sea, air) terminals each of which transmitted relatively short messages over the channel intermittently. The system was called the *Tactical Transmission System* (TATS), and it is described in a paper by Drouilhet and Bernstein (1969).

An octal Reed–Solomon (7, 2) code is used in the TATS system. Thus, two 3-bit information symbols from the input to the encoder are used to generate a seven-symbol code word. Each 3-bit coded symbol is transmitted by means of octal FSK modulation. The eight possible frequencies are spaced $1/T_c$ Hz apart, where T_c is the time (chip) duration of a single frequency transmission. In addition to the seven symbols in a code word, an eighth symbol is included. That symbol and its corresponding frequency are fixed and transmitted at the beginning of each code word for the purpose of providing timing and frequency synchronization† at the receiver. Consequently, each code word is transmitted in $8T_c$ seconds.

---

†Since mobile users are involved, there is a Doppler frequency offset associated with transmission. This frequency offset must be tracked and compensated for in the demodulation of the signal. The sync symbol is used for this purpose.

TATS was designed to transmit at information rates of 75 and 2400 bits/s. Hence, $T_c = 10$ ms and 312.5 μs, respectively. Each frequency tone corresponding to a code symbol is frequency-hopped. Hence, the hopping rate is 100 hops/s at the 75-bits/s rate and 3200 hops/s at the 2400-bits/s rate.

There are $M = 2^6 = 64$ code words in the Reed–Solomon $(7, 2)$ code and the minimum distance of the code is $d_{\min} = 6$. This means that the code provides an effective order of diversity equal to 6.

At the receiver, the received signal is first dehopped and then demodulated by passing it through a parallel bank of eight matched filters, where each filter is tuned to one of the eight possible frequencies. Each filter output is envelope-detected, quantized to 4 bits (one of 16 levels), and fed to the decoder. The decoder takes the 56 filter outputs corresponding to the reception of each seven-symbol code word and forms 64 decision variables corresponding to the 64 possible code words in the $(7, 2)$ code by linearly combining the appropriate envelope-detected outputs. A decision is made in favor of the code word having the largest decision variable.

By limiting the matched filter outputs to 16 levels, interference (crosstalk) from other users of the channel causes a relatively small loss in performance (0.75 dB with strong interference on one chip and 1.5 dB with strong interference on two chips out of the seven). The AGC used in TATS has a time constant greater than the chip interval T_c, so that no attempt is made to perform optimum weighting of the demodulator outputs as described in Section 12.3–2.

The derivation of the error probability for the TATS signal in AWGN and worst-case partial-band interference is left as an exercise for the reader (Problems 12.23 and 12.24).

12.4

OTHER TYPES OF SPREAD SPECTRUM SIGNALS

DS and FH are the most common forms of spread spectrum signals used in practice. However, other methods may be used to introduce pseudorandomness in a spread spectrum signal. One method, which is analogous to FH, is *time hopping* (TH). In TH, a time interval, which is selected to be much larger than the reciprocal of the information rate, is subdivided into a large number of time slots. The coded information symbols are transmitted in a pseudorandomly selected time slot as a block of one or more codewords. PSK modulation may be used to transmit the coded bits.

For example, suppose that a time interval T is subdivided into 1000 time slots of width $T/1000$ each. With an information bit rate of R bits/s, the number of bits to be transmitted in T seconds is RT. Coding increases this number to RT/R_c bits, where R_c is the code rate. Consequently, in a time interval of $T/1000$s, we must transmit RT/R_c bits. If binary PSK is used as the modulation method, the bit rate is $1000R/R_c$ and the bandwidth required is approximately $W = 1000R/R_c$.

A block diagram of a transmitter and a receiver for a TH spread spectrum system is shown in Figure 12.4–1. Because of the burst characteristics of the transmitted signal, buffer storage must be provided at the transmitter in a TH system, as shown in

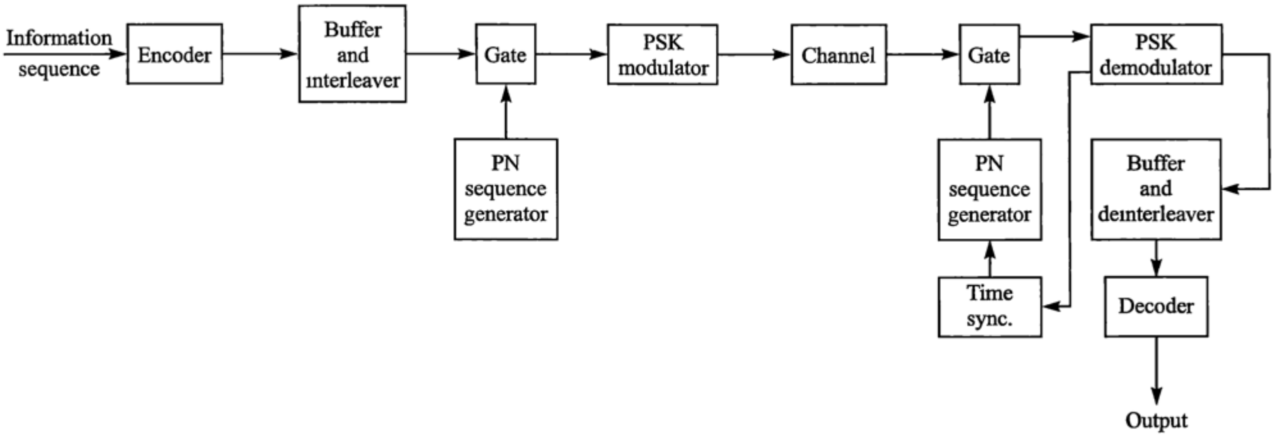

Information sequence
Encoder
Buffer and interleaver
Gate
PN sequence generator
PSK modulator
Channel
Gate
PN sequence generator
Time sync.
PSK demodulator
Buffer and deinterleaver
Decoder
Output

FIGURE 12.4–1
Block diagram of time-hopping (TH) spread spectrum system.

Figure 12.4–1. A buffer may also be used at the receiver to provide a uniform data stream to the user.

Just as partial-band interference degrades an uncoded FH spread spectrum system, partial-time (pulsed) interference has a similar effect on a TH spread spectrum system. Coding and interleaving are effective means for combating this type of interference, as we have already demonstrated for FH and DS systems. Perhaps the major disadvantage of a TH system is the stringent timing requirements compared not only with FH but, also, with DS.

Other types of spread spectrum signals can be obtained by combining DS, FH, and TH. For example, we may have a hybrid DS/FH, which means that a PN sequence is used in combination with frequency hopping. The signal transmitted on a single hop consists of a DS spread spectrum signal which is demodulated coherently. However, the received signals from different hops are combined noncoherently (envelope or square-law combining). Since coherent detection is performed within a hop, there is an advantage obtained relative to a pure FH system. However, the price paid for the gain in performance is an increase in complexity, greater cost, and more stringent timing requirements.

Another possible hybrid spread spectrum signal is DS/TH. This does not seem to be as practical as DS/FH, primarily because of an increase in system complexity and more stringent timing requirements.

12.5
SYNCHRONIZATION OF SPREAD SPECTRUM SYSTEMS

Time synchronization of the receiver to the received spread spectrum signal may be separated into two phases. There is an initial acquisition phase and a tracking phase after the signal has been initially acquired.

Acquisition In a direct sequence spread spectrum system, the PN code must be time-synchronized to within a small fraction of the chip interval $T_c \approx 1/W$. The problem of initial synchronization may be viewed as one in which we attempt to synchronize in time the receiver clock to the transmitter clock. Usually, extremely accurate and stable time clocks are used in spread spectrum systems. Consequently, accurate time clocks result in a reduction of the time uncertainty between the receiver and the transmitter. However, there is always an initial timing uncertainty due to range uncertainty between the transmitter and the receiver. This is especially a problem when communication is taking place between two mobile users. In any case, the usual procedure for establishing initial synchronization is for the transmitter to send a known pseudorandom data sequence to the receiver. The receiver is continuously in a search mode looking for this sequence in order to establish initial synchronization.

Let us suppose that the initial timing uncertainty is T_u and the chip duration is T_c. If initial synchronization is to take place in the presence of additive noise and other interference, it is necessary to dwell for $T_d = NT_c$ in order to test synchronism at each time instant. If we search over the time uncertainty interval in (coarse) time steps of $\frac{1}{2}T_c$, then the time required to establish initial synchronization is

$$T_{\text{init sync}} = \frac{T_u}{\frac{1}{2}T_c} NT_c = 2NT_u \qquad (12.5\text{–}1)$$

Clearly, the synchronization sequence transmitted to the receiver must be at least as long as $2NT_u$ in order for the receiver to have sufficient time to perform the necessary search in a serial fashion.

In principle, matched filtering or cross correlation are optimum methods for establishing initial synchronization. A filter matched to the known data waveform generated from the known pseudorandom sequence continuously looks for exceedence of a predetermined threshold. When this occurs, initial synchronization is established and the demodulator enters the "data receive" mode.

Alternatively, we may use a *sliding correlator* as shown in Figure 12.5–1. The correlator cycles through the time uncertainty, usually in discrete time intervals of $\frac{1}{2}T_c$, and correlates the received signal with the known synchronization sequence. The cross correlation is performed over the time interval NT_c (N chips) and the correlator output is compared with a threshold to determine if the known signal sequence is present. If the threshold is not exceeded, the known reference sequence is advanced in time by

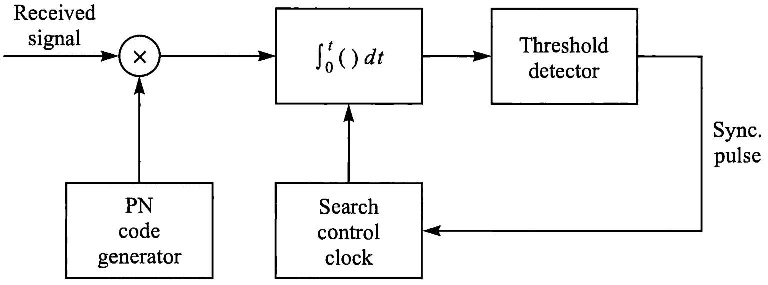


FIGURE 12.5–1
A sliding correlator for DS signal acquisition.

$\frac{1}{2}T_c$ seconds and the correlation process is repeated. These operations are performed until a signal is detected or until the search has been performed over the time uncertainty interval T_u. In the latter case, the search process is then repeated.

A similar process may also be used for FH signals. In this case, the problem is to synchronize the PN code that controls the hopped frequency pattern. To accomplish this initial synchronization, a known FH signal is transmitted to the receiver. The initial acquisition system at the receiver looks for this known FH signal pattern. For example, a bank of matched filters tuned to the transmitted frequencies in the known pattern may be employed. Their outputs must be properly delayed, envelope- or square-law-detected, weighted, if necessary, and added (noncoherent integration) to produce the signal output which is compared with a threshold. A signal present is declared when the threshold is exceeded. The search process is usually performed continuously in time until a threshold is exceeded. A block diagram illustrating this signal acquisition scheme is given in Figure 12.5–2. As an alternative, a single matched-filter–envelope detector pair may be used, preceded by an FH pattern generator and followed by a postdetection integrator and a threshold detector. This configuration, shown in Figure 12.5–3, is based on a serial search and is akin to the sliding correlator for DS spread spectrum signals.

The sliding correlator for the DS signals or its counterpart shown in Figure 12.5–3 for FH signals basically perform a serial search that is generally time-consuming. As an alternative, one may introduce some degree of parallelism by having two or more such correlators operating in parallel and searching over non-overlapping time slots. In such a case, the search time is reduced at the expense of a more complex and costly implementation.

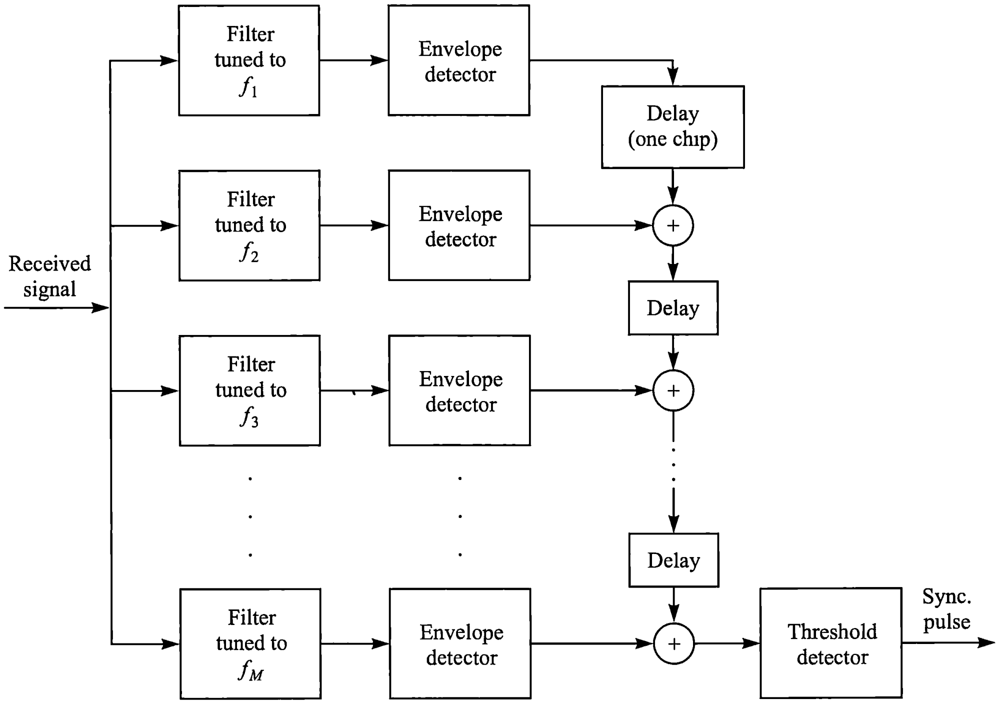

Filter tuned to f_1
Envelope detector
Delay (one chip)
Filter tuned to f_2
Envelope detector
+
Delay
Received signal
Filter tuned to f_3
Envelope detector
+
Delay
Filter tuned to f_M
Envelope detector
+
Threshold detector
Sync. pulse

FIGURE 12.5–2
System for acquisition of an FH signal.

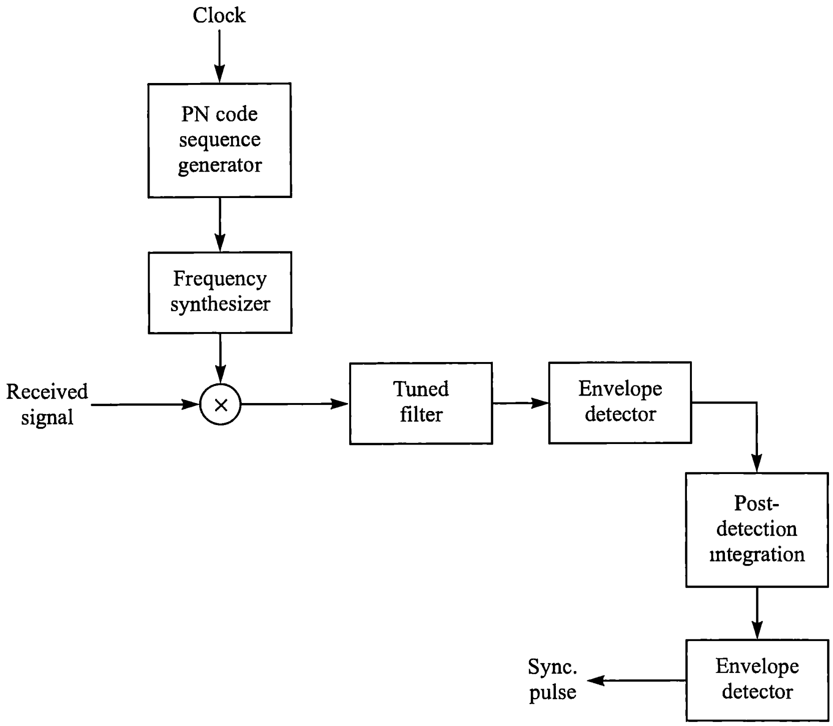


FIGURE 12.5–3
Alternative system for acquisition of an FH signal.

During the search mode, there may be false alarms that occur at the designed false alarm rate of the system. To handle the occasional false alarms, it is necessary to have an additional method or circuit that checks to confirm that the received signal at the output of the correlator remains above the threshold. With such a detection strategy, a large noise pulse that causes a false alarm will cause only a temporary exceedence of the threshold. On the other hand, when a signal is present, the correlator or matched filter output will stay above the threshold for the duration of the transmitted signal. Thus, if confirmation fails, the search is resumed.

Another initial search strategy, called a *sequential search,* has been investigated by Ward (1965) and Ward and Yiu (1977). In this method, the dwell time at each delay in the search process is made variable by employing a correlator with a variable integration period whose (biased) output is compared with two thresholds. Thus, there are three possible decisions:

1. If the upper threshold is exceeded by the correlator output, initial synchronization is declared established.
2. If the correlator output falls below the lower threshold, the signal is declared absent at that delay and the search process resumes at a different delay.
3. If the correlator output falls between the two thresholds, the integration time is increased by one chip and the resulting output is compared with the two thresholds again.

Hence, steps 1, 2, and 3 are repeated for each chip interval until the correlator output either exceeds the upper threshold or falls below the lower threshold.

The sequential search method falls in the class of sequential estimation methods proposed by Wald (1947), which are known to result in a more efficient search in the sense that the average search time is minimized. Hence, the search time for a sequential search is less than that for the fixed dwell time integrator.

In the above discussion, we have considered only time uncertainty in establishing initial synchronization. However, another aspect of initial synchronization is frequency uncertainty. If the transmitter and/or the receiver are mobile, the relative velocity between them results in a Doppler frequency shift in the received signal relative to the transmitted signal. Since the receiver does not usually know the relative velocity, a priori, the Doppler frequency shift is unknown and must be determined by means of a frequency search method. Such a search is usually accomplished in parallel over a suitably quantized frequency uncertainty interval and serially over the time uncertainty interval. A block diagram of this scheme is shown in Figure 12.5–4. Appropriate Doppler frequency search methods can also be devised for FH signals.

Tracking Once the signal is acquired, the initial search process is stopped and fine synchronization and tracking begins. The tracking maintains the PN code generator at the receiver in synchronism with the incoming signal. Tracking includes both fine chip synchronization and, for coherent demodulation, carrier phase tracking.

The commonly used tracking loop for a DS spread spectrum signal is the delay-locked loop (DLL) which is shown in Figure 12.5–5. In this tracking loop, the received signal is applied to two multipliers, where it is multiplied by two outputs from the local PN code generator, which are delayed relative to each other by an amount $2\delta \leq T_c$.

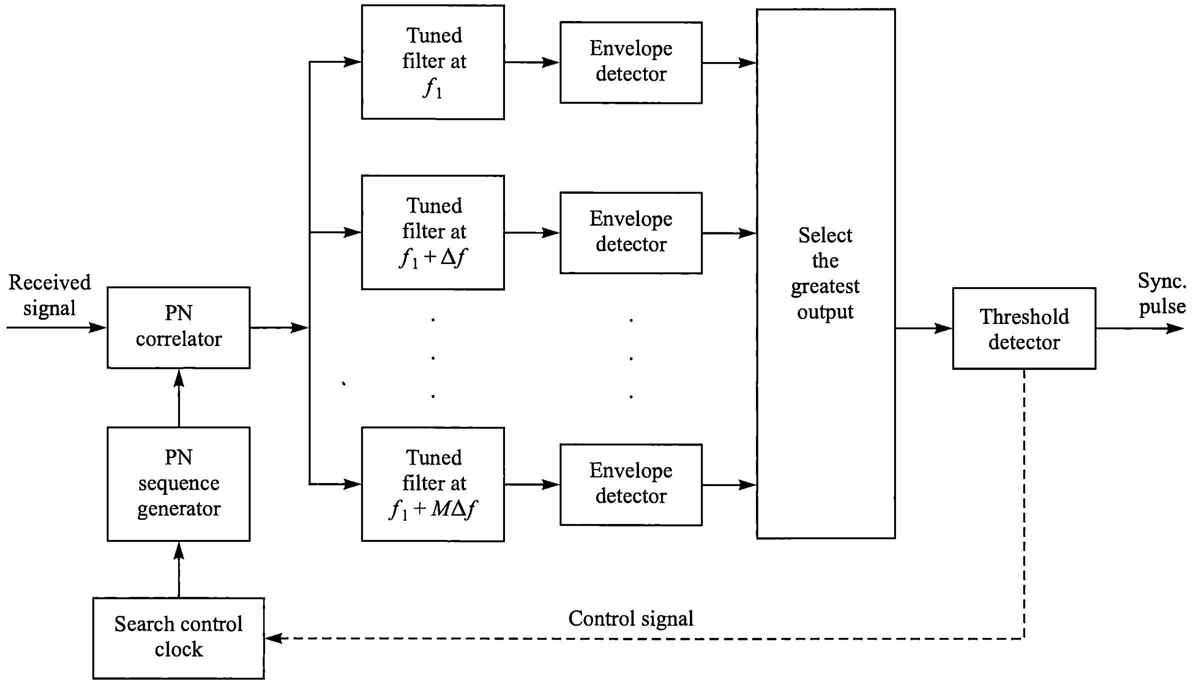

Received signal
PN correlator
PN sequence generator
Search control clock
Tuned filter at f_1
Tuned filter at $f_1+\Delta f$
Tuned filter at $f_1+M\Delta f$
Envelope detector
Envelope detector
Envelope detector
Select the greatest output
Threshold detector
Sync. pulse
Control signal

FIGURE 12.5–4
Initial search for Doppler frequency offset in a DS system.

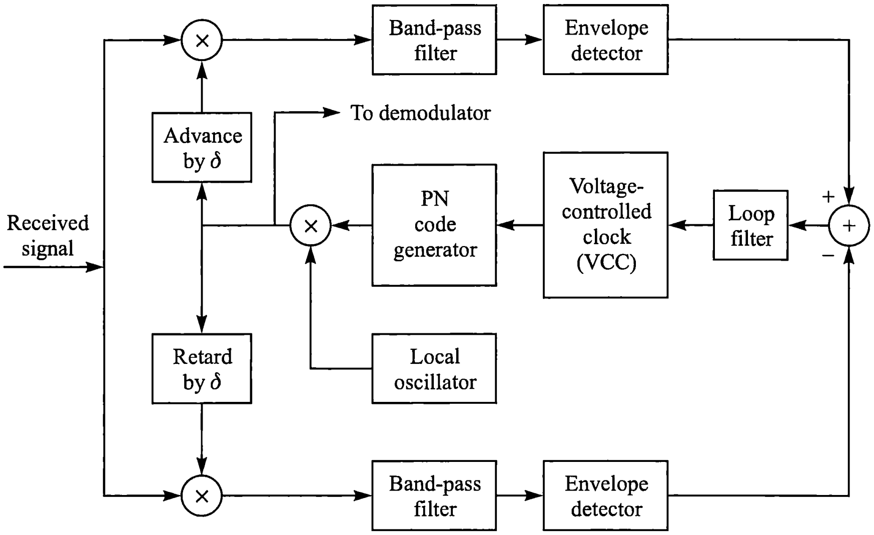

Received signal
Advance by δ
Retard by δ
Band-pass filter
Envelope detector
To demodulator
PN code generator
Voltage-controlled clock (VCC)
Loop filter
Local oscillator
Band-pass filter
Envelope detector
+
−

FIGURE 12.5–5
Delay-locked loop (DLL) for PN code tracking.

Thus, the product signals are the cross correlations between the received signal and the PN sequence at the two values of delay. These products are band-pass-filtered and envelope- (or square-law-) detected and then subtracted. This difference signal is applied to the loop filter that drives the voltage-controlled clock (VCC). The VCC serves as the clock for the PN code signal generator.

If the synchronism is not exact, the filtered output from one correlator will exceed the other and the VCC will be appropriately advanced or delayed. At the equilibrium point, the two filtered correlator outputs will be equally displaced from the peak value, and the PN code generator output will be exactly synchronized to the received signal that is fed to the demodulator. We observe that this implementation of the DLL for tracking a DS signal is equivalent to the early–late gate bit tracking synchronizer previously discussed in Section 5.3–2 and shown in Figure 5.3–5.

An alternative method for time tracking a DS signal is to use a *tau-dither loop* (TDL), illustrated by the block diagram in Figure 12.5–6. The TDL employs a single

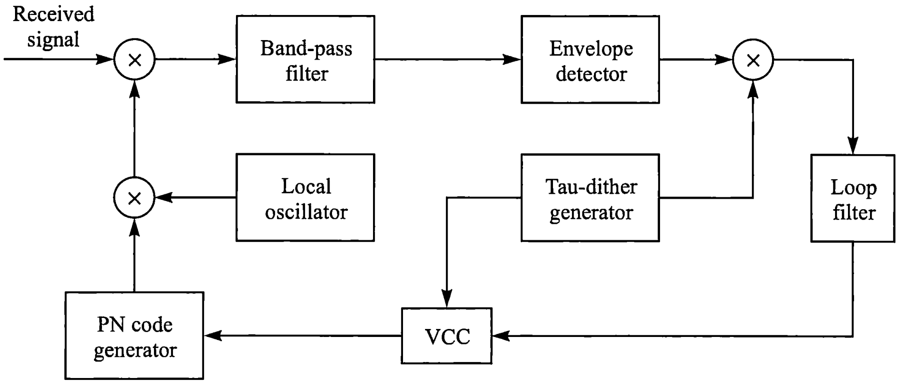

Received signal
Band-pass filter
Envelope detector
Local oscillator
Tau-dither generator
Loop filter
PN code generator
VCC

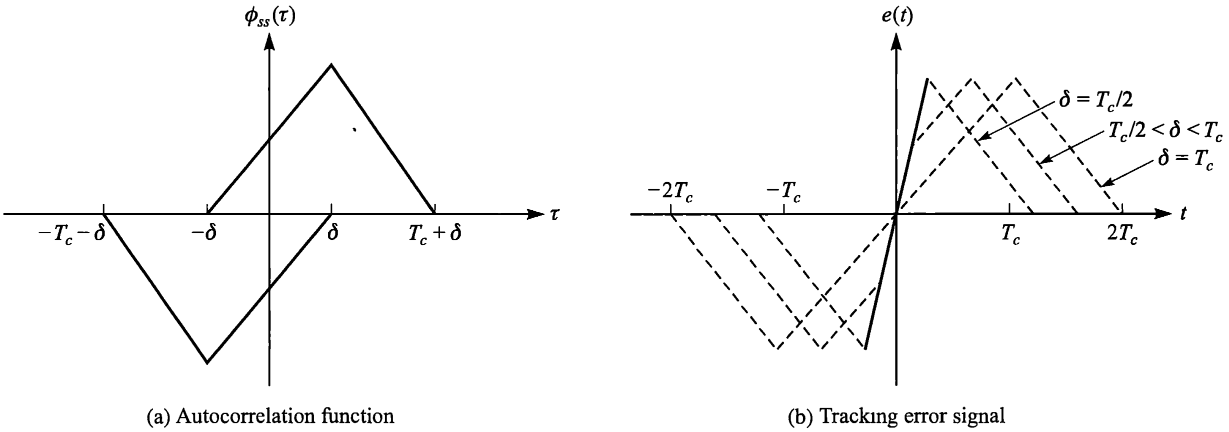
FIGURE 12.5–6
Tau-dither loop (TDL).

"arm" instead of the two "arms" shown in Figure 12.5–5. By providing a suitable gating waveform, it is possible to make this "single-arm" implementation appear to be equivalent to the "two-arm" realization. In this case, the cross correlation is regularly sampled at two values of delay, by stepping the code clock forward or backward in time by an amount δ. The envelope of the cross correlation that is sampled at $\pm\delta$ has an amplitude modulation whose phase relative to the tau-dither modulator determines the sign of the tracking error.

A major advantage of the TDL is the less costly implementation resulting from elimination of one of the two arms that are employed in the conventional DLL. A second and less apparent advantage is that the TDL does not suffer from performance degradation that is inherent in the DLL when the amplitude gain in the two arms is not properly balanced.

The DLL (and its equivalent, the TDL) generate an error signal by sampling the signal correlation function at $\pm\delta$ off the peak as shown in Figure 12.5–7a. This generates an error signal as shown in Figure 12.5–7b. The analysis of the performance of the DLL is similar to that for the phase-locked loop (PLL) carried out in Section 5.2. If it were not for the envelope detectors in the two arms of the DLL, the loop would resemble a Costas loop. In general, the variance of the time estimation error in the DLL is inversely proportional to the loop SNR, which depends on the input SNR to the loop and the loop bandwidth. Its performance is somewhat degraded as in the squaring PLL by non-linearities inherent in the envelope detectors, but this degradation is relatively small.

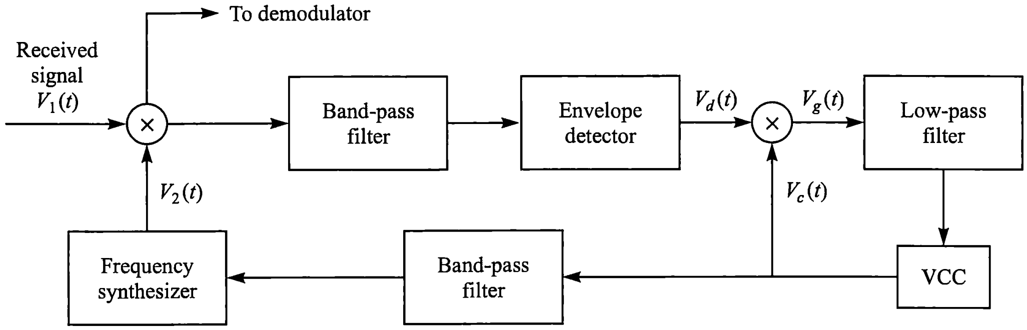
A typical tracking technique for FH spread spectrum signals is illustrated in Figure 12.5–8a. This method is also based on the premise that, although initial acquisition has been achieved, there is a small timing error between the received signal and the receiver clock. The band pass filter is tuned to a single intermediate frequency and its bandwidth is of the order of $1/T_c$, where T_c is the chip interval. Its output is envelope-detected and then multiplied by the clock signal to produce a three-level signal, as shown in Figure 12.5–8b, which drives the loop filter. Note that when the chip transitions from the locally generated sinusoidal waveform do not occur at the same time as the



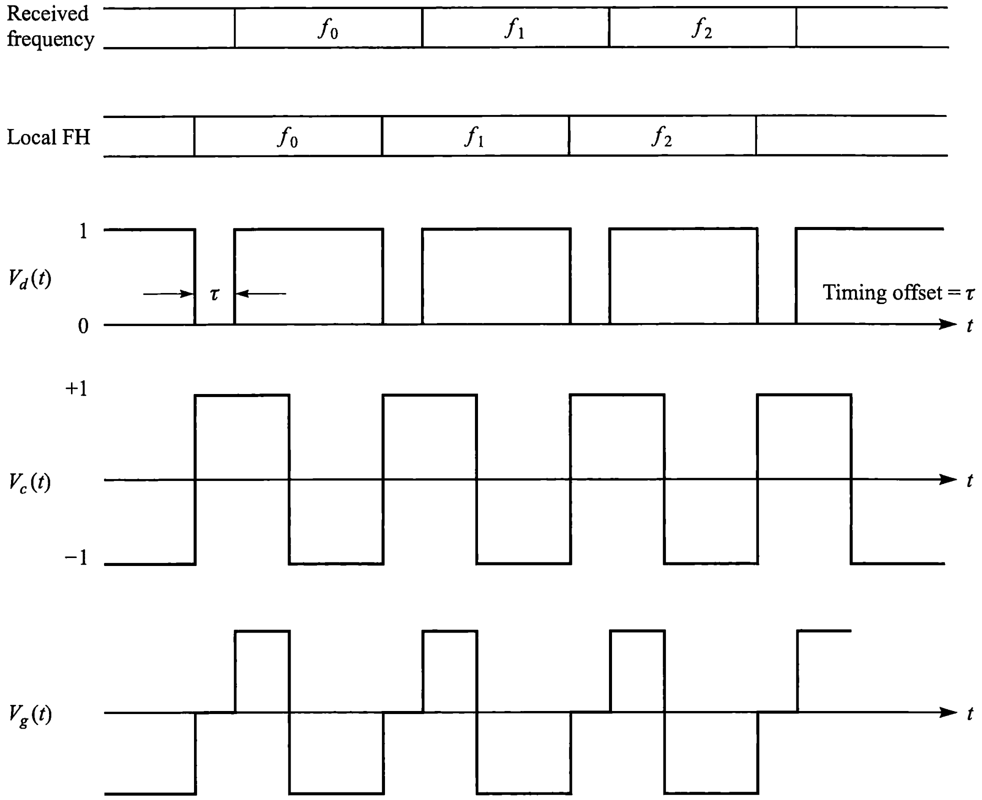
(a) Autocorrelation function

(b) Tracking error signal

FIGURE 12.5–7
Autocorrelation function and tracking error signal for DLL.


To demodulator
Received signal $V_1(t)$
Band-pass filter
Envelope detector
$V_d(t)$
$V_g(t)$
Low-pass filter
$V_2(t)$
$V_c(t)$
Frequency synthesizer
Band-pass filter
VCC

(a) Tracking loop for FH signals


Received frequency
f_0
f_1
f_2
Local FH
f_0
f_1
f_2
1
$V_d(t)$
0
τ
Timing offset = τ
t
+1
$V_c(t)$
−1
t
$V_g(t)$
t

(b) Wavefront for tracking an FH signal

FIGURE 12.5–8
Tracking method for FH signals. [*From Pickholtz et al. (1982).* © *1982 IEEE.*]

transitions in the incoming signal, the output of the loop filter will be either negative or positive, depending on whether the VCC is lagging or advanced relative to the timing of the input signal. This error signal from the loop filter will provide the control signal for adjusting the VCC timing signal so as to drive the frequency synthesized pulsed sinusoid to proper synchronism with the received signal.

12.6
BIBLIOGRAPHICAL NOTES AND REFERENCES

The introductory treatment of spread spectrum signals and their performance that we have given in this chapter is necessarily brief. Detailed and more specialized treatments of signal acquisition techniques, code tracking methods, and hybrid spread spectrum systems, as well as other general topics on spread spectrum signals and systems, can be found in the vast body of technical literature that now exists on the subject.

Historically, the primary application of spread spectrum communications has been in the development of secure (AJ) digital communication systems for military use. In fact, prior to 1970, most of the work on the design and development of spread spectrum communications was classified. Since then, this trend has been reversed. The open literature now contains numerous publications on all aspects of spread spectrum signal analysis and design. Moreover, we have recently seen the application of spread spectrum signaling techniques to commercial communications such as interoffice radio communications (see Pahlavan, 1985), mobile radio communications (see Yue, 1983), and digital cellular communications (see Viterbi, 1995).

A historical perspective on the development of spread spectrum communication systems covering the period 1920–1960 is given in a paper by Scholtz (1982). Tutorial treatments focusing on the basic concepts are found in the papers by Scholtz (1977) and Pickholtz et al. (1982). These papers also contain a large number of references to previous work. In addition, there are two papers by Viterbi (1979, 1985) that provide a basic review of the performance characteristics of DS and FH signaling techniques.

Comprehensive treatments of various aspects of analysis and design of spread spectrum signals and systems, including synchronization techniques are now available in the texts by Simon et al. (1985) Peterson et al. (1995), and Holmes (1982). In addition to these texts, there are several special issues of the *IEEE Transactions on Communications* devoted to spread spectrum communications (August 1977 and May 1982) and the *IEEE Transactions on Selected Areas in Communication* (September 1985, May 1989, May 1990, and June 1993). These issues contain a collection of papers devoted to a variety of topics, including multiple-access techniques, synchronization techniques, and performance analyses with various types of interference. A number of important papers that have been published in IEEE journals have also been reprinted in book form by the IEEE Press (Dixon, 1976; Cook et al., 1983). Finally, we recommend the book by Golomb (1967) as a basic reference on shift register sequences for the reader who wishes to delve deeper into this topic.

PROBLEMS

12.1 Following the procedure outlined in Example 12.2–2, determine the error rate performance of a DS spread spectrum system in the presence of CW jamming when the signal pulse is

$$g(t) = \sqrt{\frac{16\mathcal{E}_c}{3T_c}} \cos^2\left[\frac{\pi}{T_c}(t - \tfrac{1}{2}T_c)\right], \qquad 0 \le t \le T_c$$

12.2 The sketch in Figure P12.2 illustrates the power spectral densities of a PN spread spectrum signal and narrowband interference in an uncoded (trivial repetition code) digital communication system. Referring to Figure 12.2–6, which shows the demodulator for this signal, sketch the (approximate) spectral characteristics of the signal and the interference after the multiplication of $r(t)$ with the output of the PN generator. Determine the fraction of the total interference that appears at the output of the correlator when the number of PN chips per bit is L_c.

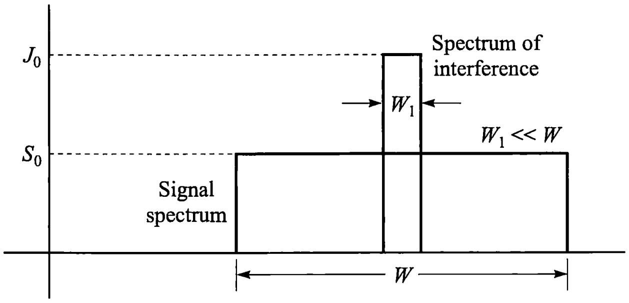


FIGURE P12.2

12.3 Consider the concatenation of a Reed–Solomon (31, 3) (q = 32-ary alphabet) as the outer code with a Hadamard (16, 5) binary code as the inner code in a DS spread spectrum system. Assume that soft-decision decoding is performed on both codes. Determine an upper (union) bound on the probability of a bit error based on the minimum distance of the concatenated code.

12.4 The Hadamard $(n, k) = (2^m, m+1)$ codes are low-rate codes with $d_{\min} = 2^{n-1}$. Determine the performance of this class of codes for DS spread spectrum signals with binary PSK modulation and either soft-decision or hard-decision decoding.

12.5 A rate 1/2 convolutional code with $d_{\text{free}} = 10$ is used to encode a data sequence occurring at a rate of 1000 bits/s. The modulation is binary PSK. The DS spread spectrum sequence has a chip rate of 10 MHz.

a. Determine the coding gain.
b. Determine the processing gain.
c. Determine the interference margin assuming an $\mathcal{E}_b/J_0 = 10$.

12.6 A total of 30 equal-power users are to share a common communication channel by CDMA. Each user transmits information at a rate of 10 kbits/s via DS spread spectrum and binary PSK. Determine the minimum chip rate to obtain a bit error probability of 10^{-5}. Additive noise at the receiver may be ignored in this computation.

12.7 A CDMA system is designed based on DS spread spectrum with a processing gain of 1000 and binary PSK modulation. Determine the number of users if each user has equal power and the desired level of performance is an error probability of 10^{-6}. Repeat the computation if the processing gain is changed to 500.

12.8 A DS spread spectrum system transmits at a rate of 1000 bits/s in the presence of a tone jammer. The jammer power is 20 dB greater than the desired signal, and the required $\mathcal{E}_b/J_0$ to achieve satisfactory performance is 10 dB.

a. Determine the spreading bandwidth required to meet the specifications.

b. If the jammer is a pulse jammer, determine the pulse duty cycle that results in worst-case jamming and the corresponding probability of error.

12.9 A CDMA system consists of 15 equal-power users that transmit information at a rate of 10,000 bits/s, each using a DS spread spectrum signal operating at a chip rate of 1 MHz. The modulation is binary PSK.

a. Determine the $\mathcal{E}_b/J_0$, where J_0 is the spectral density of the combined interference.

b. What is the processing gain?

c. How much should the processing gain be increased to allow for doubling the number of users without affecting the output SNR?

12.10 A DS binary PSK spread spectrum signal has a processing gain of 500. What is the interference margin against a continuous-tone interference if the desired error probability is 10^{-5}?

12.11 Repeat Problem 12.10 if the interference consists of pulsed noise with a duty cycle of 1 percent.

12.12 Consider the DS spread spectrum signal

$$c(t) = \sum_{n=-\infty}^{\infty} c_n p(t - nT_c)$$

where c_n is a periodic m sequence with a period $N = 127$ and $p(t)$ is a rectangular pulse of duration $T_c = 1\ \mu$s. Determine the power spectral density of the signal $c(t)$.

12.13 Suppose that $\{c_{1i}\}$ and $\{c_{2i}\}$ are two binary (0, 1) periodic sequences with periods N_1 and N_2, respectively. Determine the period of the sequence obtained by forming the modulo-2 sum of $\{c_{1i}\}$ and $\{c_{2i}\}$.

12.14 An $m = 10$ maximum-length shift register is used to generate the pseudorandom sequence in a DS spread spectrum system. The chip duration is $T_c = 1\ \mu$s, and the bit duration is $T_b = NT_c$, where N is the length (period) of the m sequence.

a. Determine the processing gain of the system in dB.

b. Determine the interference margin if the required $\mathcal{E}_b/J_0 = 10$ and the jammer is a tone jammer with an average power J_{av}.

12.15 An FH binary orthogonal FSK system employs an $m = 15$ stage linear feedback shift register that generates a maximum-length sequence. Each state of the shift register selects one of L non-overlapping frequency bands in the hopping pattern. The bit rate is 100 bits/s and the hop rate is one hop per bit. The demodulator employs noncoherent detection.

a. Determine the hopping bandwidth for this channel.

b. What is the processing gain?

c. What is the probability of error in the presence of AWGN?

12.16 Consider the FH binary orthogonal FSK system described in Problem 12.15. Suppose that the hop rate is increased to 2 hops/bit. The receiver uses square-law combining to combine the signal over the two hops.

a. Determine the hopping bandwidth for the channel.
b. What is the processing gain?
c. What is the error probability in the presence of AWGN?

12.17 In a fast FH spread spectrum system, the information is transmitted via FSK, with noncoherent detection. Suppose there are $N = 3$ hops/bit, with hard-decision decoding of the signal in each hop.

a. Determine the probability of error for this system in an AWGN channel with power spectral density $\frac{1}{2}N_0$ and an SNR $= 13$ dB (total SNR over the three hops).
b. Compare the result in (a) with the error probability of an FH spread spectrum system that hops once per bit.

12.18 A slow FH binary FSK system with noncoherent detection operates at $\mathcal{E}_b/J_0 = 10$, with a hopping bandwidth of 2 GHz, and a bit rate of 10 kbits/s.

a. What is the processing gain for the system?
b. If the jammer operates as a partial-band jammer, what is the bandwidth occupancy for worst-case jamming?
c. What is the probability of error for the worst-case partial-band jammer?

12.19 Determine the error probability for an FH spread spectrum signal in which a binary convolutional code is used in combination with binary FSK. The interference on the channel is AWGN. The FSK demodulator outputs are square-law-detected and passed to the decoder, which performs optimum soft-decision Viterbi decoding as described in Chapter 8. Assume that the hopping rate is 1 hop per coded bit.

12.20 Repeat Problem 12.19 for hard-decision Viterbi decoding.

12.21 Repeat Problem 12.19 when fast frequency hopping is performed at a hopping rate of L hops per coded bit.

12.22 Repeat Problem 12.19 when fast frequency hopping is performed with L hops per coded bit and the decoder is a hard-decision Viterbi decoder. The L chips per coded bit are square-law-detected and combined prior to the hard decision.

12.23 The TATS signal described in Section 12.3–3 is demodulated by a parallel bank of eight matched filters (octal FSK), and each filter output is square-law-detected. The eight outputs obtained in each of seven signal intervals (56 total outputs) are used to form the 64 possible decision variables corresponding to the Reed–Solomon (7, 2) code. Determine an upper (union) bound of the code word error probability for AWGN and soft-decision decoding.

12.24 Repeat Problem 12.23 for the worst-case partial-band interference channel.

12.25 Derive the results in Equations 12.2–50 and 12.2–51 from Equation 12.2–49.

12.26 Show that Equation 12.3–14 follows from Equation 12.3–13.

12.27 Derive Equation 12.3–17 from Equation 12.3–16.

12.28 The parity polynomials for constructing Gold code sequences of length $n = 7$ are

$$h_1(X) = X^3 + X + 1$$
$$h_2(X) = X^3 + X^2 + 1$$

Generate all the Gold codes of length 7 and determine the cross correlations of one sequence with each of the others.

12.29 In Section 12.2–3, we demonstrated techniques for evaluating the error probability of a coded system with interleaving in pulse interference by using the cutoff rate parameter R_0. Use the error probability curves given in Figure P12.29 for rate 1/2 and 1/3 convolutional codes with soft-decision Viterbi decoding to determine the corresponding error rates for a coded system in pulse interference. Perform this computation for $K = 3, 5$, and 7.

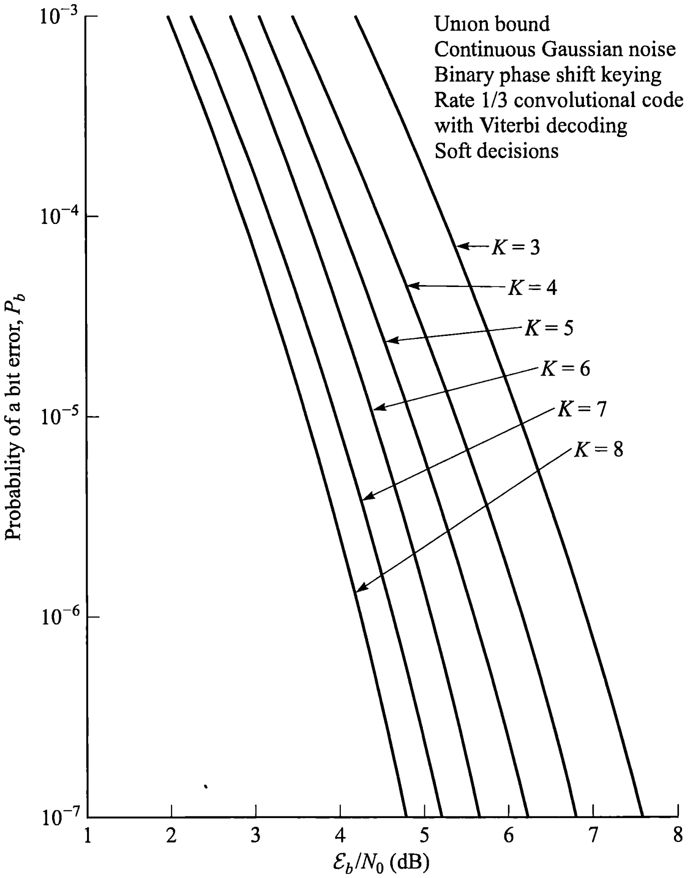


FIGURE P12.29

12.30 In coded and interleaved DS binary PSK modulation with pulse jamming and soft-decision decoding, the cutoff rate is

$$R_0 = 1 - \log_2(1 + \alpha e^{-\alpha \mathcal{E}_c/N_0})$$

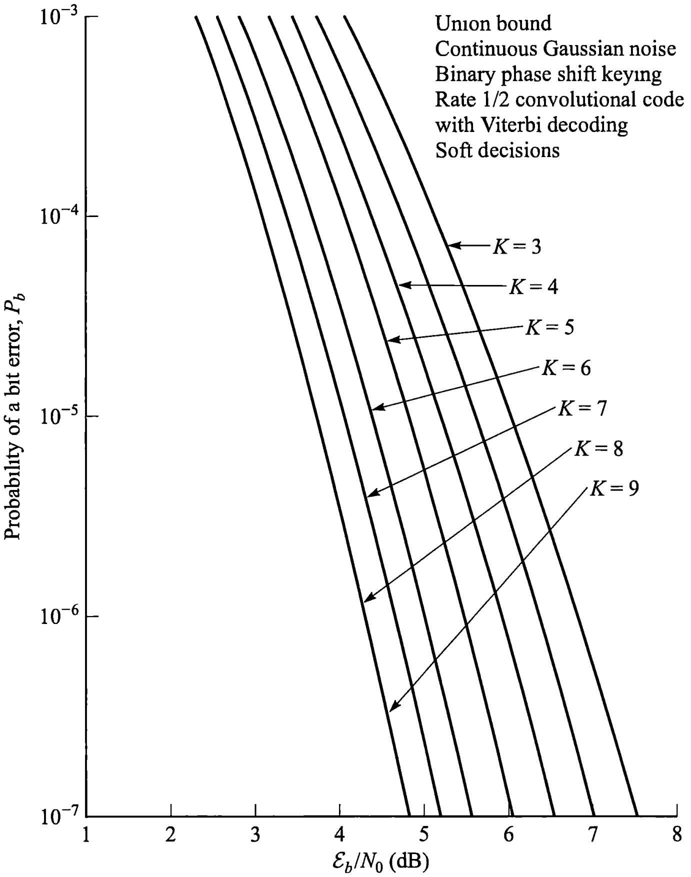


FIGURE P12.29
(Continued)

where α is the fraction of the time the system is being jammed, $\mathcal{E}_c = \mathcal{E}_b R$, R is the bit rate, and $N_0 \equiv J_0$.

a. Show that the SNR per bit, $\mathcal{E}_b/N_0$, can be expressed as

$$\frac{\mathcal{E}_b}{N_0} = \frac{1}{\alpha R} \ln \frac{\alpha}{2^{1-R_0} - 1}$$

b. Determine the value of α that maximizes the required $\mathcal{E}_b/N_0$ (worst-case pulse jamming) and the resulting maximum value of $\mathcal{E}_b/N_0$.

c. Plot the graph of $10 \log(\mathcal{E}_b/rN_0)$ versus R_0, where $r = R_0/R$, for worst-case pulse jamming and for AWGN ($\alpha = 1$). What conclusions do you reach regarding the effect of worst-case pulse jamming?

12.31 In a coded and interleaved FH q-ary FSK modulation with partial band jamming and coherent demodulation with soft-decision decoding, the cutoff rate is

$$R_0 = \log_2 \left[\frac{q}{1 + (q-1)\alpha e^{-\alpha \mathcal{E}_c / 2N_0}} \right]$$

where α is the fraction of the band being jammed, $\mathcal{E}_c$ is the chip (or tone) energy, and $N_0 = J_0$.

a. Show that the SNR per bit can be expressed as

$$\frac{\mathcal{E}_b}{N_0} = \frac{2}{\alpha R} \ln \frac{(q-1)\alpha}{q2^{-R_0} - 1}$$

b. Determine the value of α that maximizes the required $\mathcal{E}_b/N_0$ (worst-case partial band jamming) and the resulting maximum value of $\mathcal{E}_b/N_0$.

c. Define $r = R_0/R$ in the result for $\mathcal{E}_b/N_0$ from (b), and plot $10\log(\mathcal{E}_b/rN_0)$ versus the normalized cutoff rate $R_0/\log_2 q$ for $q = 2, 4, 8, 16, 32$. Compare these graphs with the results of Problem 12.30c. What conclusions do you reach regarding the effect of worst-case partial band jamming? What is the effect of increasing the alphabet size q? What is the penalty in SNR between the results in Problem 12.30c and q-ary FSK as $q \to \infty$?

13

Fading Channels I: Characterization and Signaling

The previous chapters have described the design and performance of digital communication systems for transmission on either the classical AWGN channel or a linear filter channel with AWGN. We observed that the distortion inherent in linear filter channels requires special signal design techniques and rather sophisticated adaptive equalization algorithms in order to achieve good performance.

In this chapter, we consider the signal design, receiver structure, and receiver performance for more complex channels, namely, channels having randomly time variant impulse responses. This characterization serves as a model for signal transmission over many radio channels such as shortwave ionospheric radio communication in the 3–30 MHz frequency band (HF), tropsopheric scatter (beyond-the-horizon) radio communications in the 300–3000 MHz frequency band (UHF), and 3000–30,000 MHz frequency band (SHF), and ionospheric forward scatter in the 30–300 MHz frequency band (VHF). The time-variant impulse responses of these channels are a consequence of the constantly changing physical characteristics of the media. For example, the ions in the ionospheric layers that reflect the signals transmitted in the HF band are always in motion. To the user of the channel, the motion of the ions appears to be random. Consequently, if the same signal is transmitted at HF in two widely separated time intervals, the two received signals will be different. The time-varying responses that occur are treated in statistical terms.

We shall begin our treatment of digital signaling over fading multipath channels by first developing a statistical characterization of the channel. Then we shall evaluate the performance of several basic digital signaling techniques for communication over such channels. The performance results will demonstrate the severe penalty in SNR that must be paid as a consequence of the fading characteristics of the received signal. We shall then show that the penalty in SNR can be dramatically reduced by means of efficient modulation/coding and demodulation/decoding techniques.

13.1
CHARACTERIZATION OF FADING MULTIPATH CHANNELS

If we transmit an extremely short pulse, ideally an impulse, over a time-varying multipath channel, the received signal might appear as a train of pulses, as shown in Figure 13.1–1. Hence, one characteristic of a multipath medium is the time spread introduced in the signal that is transmitted through the channel.

A second characteristic is due to the time variations in the structure of the medium. As a result of such time variations, the nature of the multipath varies with time. That is, if we repeat the pulse-sounding experiment over and over, we shall observe changes in the received pulse train, which will include changes in the sizes of the individual pulses, changes in the relative delays among the pulses, and, quite often, changes in the number of pulses observed in the received pulse train as shown in Figure 13.1–1. Moreover, the time variations appear to be unpredictable to the user of the channel. Therefore, it is reasonable to characterize the time-variant multipath channel statistically. Toward this end, let us examine the effects of the channel on a transmitted signal that is represented in general as

$$s(t) = \text{Re}\left[s_l(t)e^{j2\pi f_c t}\right] \tag{13.1–1}$$

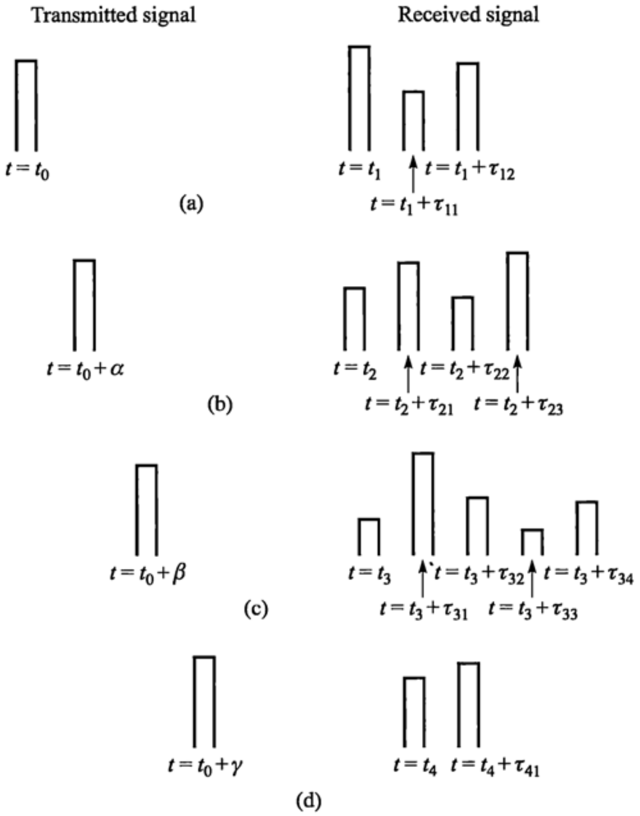

Transmitted signal
Received signal
$t=t_0$
(a)
$t=t_1$
$t=t_1+\tau_{12}$
$t=t_1+\tau_{11}$
$t=t_0+\alpha$
(b)
$t=t_2$
$t=t_2+\tau_{22}$
$t=t_2+\tau_{21}$
$t=t_2+\tau_{23}$
$t=t_0+\beta$
(c)
$t=t_3$
$t=t_3+\tau_{32}$
$t=t_3+\tau_{34}$
$t=t_3+\tau_{31}$
$t=t_3+\tau_{33}$
$t=t_0+\gamma$
(d)
$t=t_4$
$t=t_4+\tau_{41}$

FIGURE 13.1–1
Example of the response of a time-variant multipath channel to a very narrow pulse.

We assume that there are multiple propagation paths. Associated with each path is a propagation delay and an attenuation factor. Both the propagation delays and the attenuation factors are time-variant as a result of changes in the structure of the medium. Thus, the received bandpass signal may be expressed in the form

$$x(t) = \sum_n \alpha_n(t) s[t - \tau_n(t)] \tag{13.1–2}$$

where $\alpha_n(t)$ is the attenuation factor for the signal received on the nth path and $\tau_n(t)$ is the propagation delay for the nth path. Substitution for $s(t)$ from Equation 14.1–1 into Equation 13.1–2 yields the result

$$x(t) = \text{Re}\left(\left\{\sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} s_l[t - \tau_n(t)]\right\} e^{j2\pi f_c t}\right) \tag{13.1–3}$$

It is apparent from Equation 13.1–3 that in the absence of noise the equivalent lowpass received signal is

$$r_l(t) = \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} s_l[t - \tau_n(t)] \tag{13.1–4}$$

Since $r_l(t)$ is the response of an equivalent lowpass channel to the equivalent lowpass signal $s_l(t)$, it follows that the equivalent lowpass channel is described by the time-variant impulse response

$$c(\tau; t) = \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} \delta[\tau - \tau_n(t)] \tag{13.1–5}$$

For some channels, such as the tropospheric scatter channel, it is more appropriate to view the received signal as consisting of a continuum of multipath components. In such a case, the received signal $x(t)$ is expressed in the integral form

$$x(t) = \int_{-\infty}^{\infty} \alpha(\tau; t) s(t - \tau) d\tau \tag{13.1–6}$$

where $\alpha(\tau; t)$ denotes the attenuation of the signal components at delay τ and at time instant t. Now substitution for $s(t)$ from Equation 13.1–1 into Equation 13.1–6 yields

$$x(t) = \text{Re}\left\{\left[\int_{-\infty}^{\infty} \alpha(\tau; t) e^{-j2\pi f_c \tau} s_l(t - \tau)\, d\tau\right] e^{j2\pi f_c t}\right\} \tag{13.1–7}$$

Since the integral in Equation 13.1–7 represents the convolution of $s_l(t)$ with an equivalent lowpass time-variant impulse response $c(\tau; t)$, it follows that

$$c(\tau; t) = \alpha(\tau; t) e^{-j2\pi f_c \tau} \tag{13.1–8}$$

where $c(\tau; t)$ represents the response of the channel at time t due to an impulse applied at time $t - \tau$. Thus Equation 13.1–8 is the appropriate definition of the equivalent lowpass impulse response when the channel results in continuous multipath and Equation 13.1–5 is appropriate for a channel that contains discrete multipath components.

Now let us consider the transmission of an unmodulated carrier at frequency f_c. Then $s_l(t) = 1$ for all t, and, hence, the received signal for the case of discrete multipath, given by Equation 13.1–4, reduces to

$$\begin{aligned} r_l(t) &= \sum_n \alpha_n(t) e^{-j2\pi f_c \tau_n(t)} \\ &= \sum_n \alpha_n(t) e^{j\theta_n(t)} \end{aligned} \tag{13.1–9}$$

where $\theta_n(t) = -2\pi f_c \tau_n(t)$. Thus, the received signal consists of the sum of a number of time-variant vectors (phasors) having amplitudes $\alpha_n(t)$ and phases $\theta_n(t)$. Note that large dynamic changes in the medium are required for $\alpha_n(t)$ to change sufficiently to cause a significant change in the received signal. On the other hand, $\theta_n(t)$ will change by 2π rad whenever τ_n changes by $1/f_c$. But $1/f_c$ is a small number and, hence, θ_n can change by 2π rad with relatively small motions of the medium. We also expect the delays $\tau_n(t)$ associated with the different signal paths to change at different rates and in an unpredictable (random) manner. This implies that the received signal $r_l(t)$ in Equation 13.1–9 can be modeled as a random process. When there are a large number of paths, the central limit theorem can be applied. That is, $r_l(t)$ may be modeled as a complex-valued Gaussian random process. This means that the time-variant impulse response $c(\tau; t)$ is a complex-valued Gaussian random process in the t variable.

The multipath propagation model for the channel embodied in the received signal $r_l(t)$, given in Equation 13.1–9, results in signal fading. The fading phenomenon is primarily a result of the time variations in the phases $\{\theta_n(t)\}$. That is, the randomly time variant phases $\{\theta_n(t)\}$ associated with the vectors $\{\alpha_n e^{j\theta_n}\}$ at times result in the vectors adding destructively. When that occurs, the resultant received signal $r_l(t)$ is very small or practically zero. At other times, the vectors $\{\alpha_n e^{j\theta_n}\}$ add constructively, so that the received signal is large. Thus, the amplitude variations in the received signal, termed *signal fading*, are due to the time-variant multipath characteristics of the channel.

When the impulse response $c(\tau; t)$ is modeled as a zero-mean complex-valued Gaussian process, the envelope $|c(\tau; t)|$ at any instant t is Rayleigh-distributed. In this case the channel is said to be a *Rayleigh fading channel*. In the event that there are fixed scatterers or signal reflectors in the medium, in addition to randomly moving scatterers, $c(\tau; t)$ can no longer be modeled as having zero-mean. In this case, the envelope $|c(\tau; t)|$ has a Rice distribution and the channel is said to be a *Ricean fading channel*. Another probability distribution function that has been used to model the envelope of fading signals is the Nakagami-*m* distribution. These fading channel models are considered in Section 13.1–2.

13.1–1 Channel Correlation Functions and Power Spectra

We shall now develop a number of useful correlation functions and power spectral density functions that define the characteristics of a fading multipath channel. Our starting point is the equivalent lowpass impulse response $c(\tau; t)$, which is characterized as a complex-valued random process in the t variable. We assume that $c(\tau; t)$ is wide-sense-stationary. Then we define the autocorrelation function of $c(\tau; t)$ as

$$R_c(\tau_2, \tau_1; \Delta t) = E\left[c^*(\tau_1; t) c(\tau_2; t + \Delta t)\right] \tag{13.1–10}$$

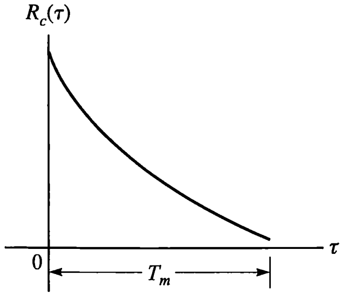


FIGURE 13.1–2
Multipath intensity profile.

In most radio transmission media, the attentuation and phase shift of the channel associated with path delay τ_1 is uncorrelated with the attenuation and phase shift associated with path delay τ_2. This is usually called *uncorrelated scattering*. We make the assumption that the scattering at two different delays is uncorrelated and incorporate it into Equation 13.1–10 to obtain

$$E\left[c^*(\tau_1; t)c(\tau_2; t + \Delta t)\right] = R_c(\tau_1; \Delta t)\delta(\tau_2 - \tau_1) \qquad (13.1\text{–}11)$$

If we let $\Delta t = 0$, the resulting autocorrelation function $R_c(\tau; 0) \equiv R_c(\tau)$ is simply the average power output of the channel as a function of the time delay τ. For this reason, $R_c(\tau)$ is called the *multipath intensity profile* or the *delay power spectrum* of the channel. In general, $R_c(\tau; \Delta t)$ gives the average power output as a function of the time delay τ and the difference Δt in observation time.

In practice, the function $R_c(\tau; \Delta t)$ is measured by transmitting very narrow pulses or, equivalently, a wideband signal and cross-correlating the received signal with a delayed version of itself. Typically, the measured function $R_c(\tau)$ may appear as shown in Figure 13.1–2. The range of values of τ over which $R_c(\tau)$ is essentially nonzero is called the *multipath spread of the channel* and is denoted by T_m.

A completely analogous characterization of the time-variant multipath channel begins in the frequency domain. By taking the Fourier transform of $c(\tau; t)$, we obtain the time-variant transfer function $C(f; t)$, where f is the frequency variable. Thus,

$$C(f; t) = \int_{-\infty}^{\infty} c(\tau; t)e^{-j2\pi f\tau}\, d\tau \qquad (13.1\text{–}12)$$

If $c(\tau; t)$ is modeled as a complex-valued zero-mean Gaussian random process in the t variable, it follows that $C(f; t)$ also has the same statistics. Under the assumption that the channel is wide-sense-stationary, we define the autocorrelation function

$$R_C(f_2, f_1; \Delta t) = E\left[C^*(f_1; t)C(f_2; t + \Delta t)\right] \qquad (13.1\text{–}13)$$

Since $C(f; t)$ is the Fourier transform of $c(\tau; t)$, it is not surprising to find that $R_C(f_2, f_1; \Delta t)$ is related to $R_c(\tau; \Delta t)$ by the Fourier transform. The relationship is easily established by substituting Equation 13.1–12 into Equation 13.1–13. Thus,

$$\begin{aligned}
R_C(f_2, f_1; \Delta t) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} E\left[c^*(\tau_1; t)c(\tau_2; t+\Delta t)\right] e^{j2\pi(f_1\tau_1 - f_2\tau_2)} d\tau_1 d\tau_2 \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} R_c(\tau_1; \Delta t)\delta(\tau_2 - \tau_1) e^{j2\pi(f_1\tau_1 - f_2\tau_2)} d\tau_1 d\tau_2 \\
&= \int_{-\infty}^{\infty} R_c(\tau_1; \Delta t) e^{j2\pi(f_1 - f_2)\tau_1} d\tau_1 \\
&= \int_{-\infty}^{\infty} R_c(\tau_1; \Delta t) e^{-j2\pi\Delta f\,\tau_1} d\tau_1 \equiv R_C(\Delta f; \Delta t) \qquad (13.1\text{–}14)
\end{aligned}$$

where $\Delta f = f_2 - f_1$. From Equation 13.1–14, we observe that $R_C(\Delta f; \Delta t)$ is the Fourier transform of the multipath intensity profile. Furthermore, the assumption of uncorrelated scattering implies that the autocorrelation function of $C(f; t)$ in frequency is a function of only the frequency difference $\Delta f = f_2 - f_1$. Therefore, it is appropriate to call $R_C(\Delta f; \Delta t)$ the *spaced-frequency, spaced time correlation function of the channel.* It can be measured in practice by transmitting a pair of sinusoids separated by Δf and cross-correlating the two separately received signals with a relative delay Δt.

Suppose we set $\Delta t = 0$ in Equation 13.1–14. Then, with $R_C(\Delta f; 0) \equiv R_C(\Delta f)$ and $R_c(\tau; 0) \equiv R_c(\tau)$, the transform relationship is simply

$$R_C(\Delta f) = \int_{-\infty}^{\infty} R_c(\tau) e^{-j2\pi\Delta f\,\tau} d\tau \qquad (13.1\text{–}15)$$

The relationship is depicted graphically in Figure 13.1–3. Since $R_C(\Delta f)$ is an autocorrelation function in the frequency variable, it provides us with a measure of the frequency coherence of the channel. As a result of the Fourier transform relationship between $R_C(\Delta f)$ and $R_c(\tau)$, the reciprocal of the multipath spread is a measure of the *coherence bandwidth of the channel.* That is,

$$(\Delta f)_c \approx \frac{1}{T_m} \qquad (13.1\text{–}16)$$

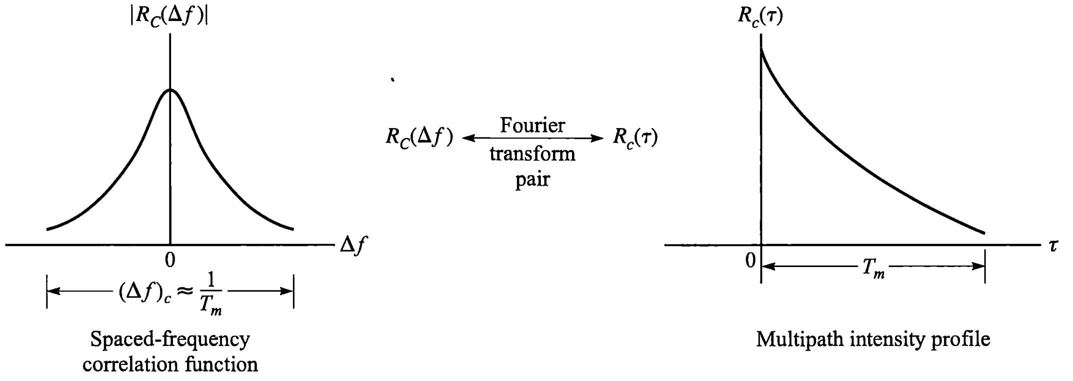


FIGURE 13.1–3
Relationship between $R_C(\Delta f)$ and $R_c(\tau)$.

where $(\Delta f)_c$ denotes the coherence bandwidth. Thus, two sinusoids with frequency separation greater than $(\Delta f)_c$ are affected differently by the channel. When an information-bearing signal is transmitted through the channel, if $(\Delta f)_c$ is small in comparison to the bandwidth of the transmitted signal, the channel is said to be *frequency-selective.* In this case, the signal is severely distorted by the channel. On the other hand, if $(\Delta f)_c$ is large in comparison with the bandwidth of the transmitted signal, the channel is said to be *frequency-nonselective.*

We now focus our attention on the time variations of the channel as measured by the parameter Δt in $R_C(\Delta f;\, \Delta t)$. The time variations in the channel are evidenced as a Doppler broadening and, perhaps, in addition as a Doppler shift of a spectral line. In order to relate the Doppler effects to the time variations of the channel, we define the Fourier transform of $R_C(\Delta f;\, \Delta t)$ with respect to the variable Δt to be the function $\mathcal{S}_C(\Delta f;\, \lambda)$. That is,

$$\mathcal{S}_C(\Delta f;\lambda) = \int_{-\infty}^{\infty} R_C(\Delta f;\, \Delta t)e^{-j2\pi\lambda\,\Delta t}d\Delta t \tag{13.1–17}$$

With Δf set to zero and $\mathcal{S}_C(0;\, \lambda) \equiv \mathcal{S}_C(\lambda)$, the relation in Equation 14.1–17 becomes

$$\mathcal{S}_C(\lambda) = \int_{-\infty}^{\infty} R_C(0;\, \Delta t)e^{-j2\pi\lambda\,\Delta t}d\Delta t \tag{13.1–18}$$

The function $\mathcal{S}_C(\lambda)$ is a power spectrum that gives the signal intensity as a function of the Doppler frequency λ. Hence, we call $\mathcal{S}_C(\lambda)$ the *Doppler power spectrum of the channel.*

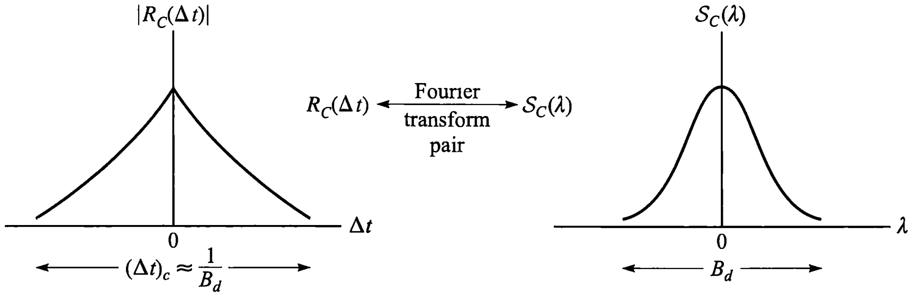
From Equation 13.1–18, we observe that if the channel is time-invariant, $R_C(\Delta t) = 1$ and $\mathcal{S}_C(\lambda)$ becomes equal to the delta function $\delta(\lambda)$. Therefore, when there are no time variations in the channel, there is no spectral broadening observed in the transmission of a pure frequency tone.

The range of values of λ over which $\mathcal{S}_C(\lambda)$ is essentially nonzero is called the *Doppler spread B_d of the channel.* Since $\mathcal{S}_C(\lambda)$ is related to $R_C(\Delta t)$ by the Fourier transform, the reciprocal of B_d is a measure of the coherence time of the channel. That is,

$$(\Delta t)_c \approx \frac{1}{B_d} \tag{13.1–19}$$

where $(\Delta t)_c$ denotes the *coherence time.* Clearly, a slowly changing channel has a large coherence time or, equivalently, a small Doppler spread. Figure 13.1–4 illustrates the relationship between $R_C(\Delta t)$ and $\mathcal{S}_C(\lambda)$.

We have now established a Fourier transform relationship between $R_C(\Delta f;\, \Delta t)$ and $R_c(\tau;\, \Delta t)$ involving the variables $(\tau, \Delta f)$, and a Fourier transform relationship between $R_C(\Delta f;\, \Delta t)$ and $\mathcal{S}_C(\Delta f;\, \lambda)$ involving the variables $(\Delta t, \lambda)$. There are two additional Fourier transform relationships that we can define, which serve to relate $R_c(\tau;\, \Delta t)$ to $\mathcal{S}_C(\Delta f;\, \lambda)$ and, thus, close the loop. The desired relationship is obtained by defining a new function, denoted by $\mathcal{S}(\tau;\, \lambda)$, to be the Fourier transform of $R_c(\tau;\, \Delta t)$


$|R_C(\Delta t)|$
$R_C(\Delta t) \xleftrightarrow[\text{transform pair}]{\text{Fourier}} \mathcal{S}_C(\lambda)$
$\mathcal{S}_C(\lambda)$
0
Δt
$(\Delta t)_c \approx \frac{1}{B_d}$
0
B_d
λ

Spaced-time correlation fnction

Doppler power spectrum

FIGURE 13.1–4
Relationship between $R_C(\Delta t)$ and $\mathcal{S}_C(\lambda)$.

in the Δt variable. That is,

$$\mathcal{S}(\tau;\lambda) = \int_{-\infty}^{\infty} R_c(\tau;\Delta t)e^{-j2\pi\lambda\,\Delta t}\,d\Delta t \tag{13.1–20}$$

It follows that $\mathcal{S}(\tau;\lambda)$ and $\mathcal{S}_C(\Delta f;\lambda)$ are a Fourier transform pair. That is,

$$\mathcal{S}(\tau;\lambda) = \int_{-\infty}^{\infty} \mathcal{S}_C(\Delta f;\lambda)e^{j2\pi\tau\,\Delta f}\,d\Delta f \tag{13.1–21}$$

Furthermore, $\mathcal{S}(\tau;\lambda)$ and $R_C(\Delta f;\Delta t)$ are related by the double Fourier transform

$$\mathcal{S}(\tau;\lambda) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} R_C(\Delta f;\Delta t)e^{-j2\pi\lambda\,\Delta t}e^{j2\pi\tau\,\Delta f}\,d\Delta t\,d\Delta f \tag{13.1–22}$$

This new function $\mathcal{S}(\tau;\lambda)$ is called the *scattering function of the channel.* It provides us with a measure of the average power output of the channel as a function of the time delay τ and the Doppler frequency λ.

The relationships among the four functions $R_C(\Delta f;\Delta t)$, $R_c(\tau;\Delta t)$, $\mathcal{S}_C(\Delta f;\lambda)$, and $\mathcal{S}(\tau;\lambda)$ are summarized in Figure 13.1–5.

EXAMPLE 13.1–1. SCATTERING FUNCTION OF A TROPOSPHERIC SCATTER CHANNEL. The scattering function $\mathcal{S}(\tau;\lambda)$ measured on a 150-mi tropospheric scatter link is shown in Figure 13.1–6. The signal used to probe the channel had a time resolution of 0.1 μs. Hence, the time-delay axis is quantized in increments of 0.1 μs. From the graph, we observe that the multipath spread $T_m = 0.7\,\mu$s. On the other hand, the Doppler spread, which may be defined as the 3-dB bandwidth of the power spectrum for each signal path, appears to vary with each signal path. For example, in one path it is less than 1 Hz, while in some other paths it is several hertz. For our purposes, we shall take the largest of these 3-dB bandwidths of the various paths and call that the *Doppler spread.*

EXAMPLE 13.1–2. MULTIPATH INTENSITY PROFILE OF MOBILE RADIO CHANNELS. The multipath intensity profile of a mobile radio channel depends critically on the type of terrain. Numerous measurements have been made under various conditions in many parts of the world. In urban and suburban areas, typical values of multipath spreads

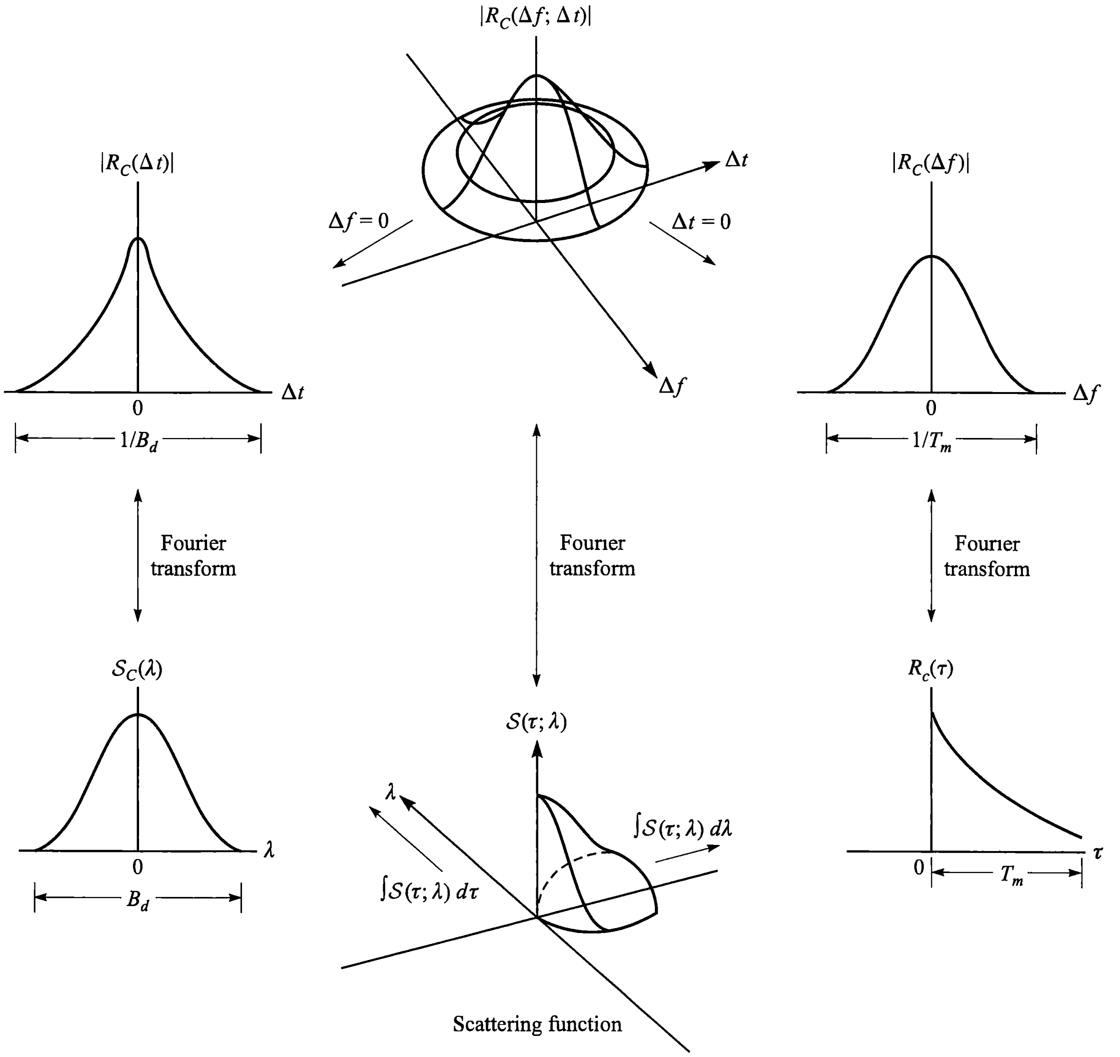

$|R_C(\Delta f; \Delta t)|$
$\Delta f = 0$
Δt
$\Delta t = 0$
Δf
$|R_C(\Delta t)|$
Δt
0
$1/B_d$
$|R_C(\Delta f)|$
Δf
0
$1/T_m$
Fourier transform
Fourier transform
Fourier transform
$S_C(\lambda)$
λ
0
B_d
$S(\tau; \lambda)$
λ
$\int S(\tau; \lambda)\, d\tau$
$\int S(\tau; \lambda)\, d\lambda$
Scattering function
$R_c(\tau)$
τ
0
T_m

FIGURE 13.1-5

Relationships among the channel correlation functions and power spectra. [*From Green (1962), with permission.*]

range from 1 to 10 μs. In rural mountainous areas, the multipath spreads are much greater, with typical values in the range of 10 to 30 μs. Two models for the multipath intensity profile that are widely used in evaluating system performance for these two types of terrain are illustrated in Figure 13.1–7.

EXAMPLE 13.1-3. DOPPLER POWER SPECTRUM OF MOBILE RADIO CHANNELS. A widely used model for the Doppler power spectrum of a mobile radio channel is the so-called Jakes' model (Jakes, 1974). In this model, the autocorrelation of the time-variant transfer function $C(f; t)$ is given as

$$\begin{aligned} R_C(\Delta t) &= E[C^*(f; t)C(f; t + \Delta t)] \\ &= J_0(2\pi f_m \Delta t) \end{aligned}$$

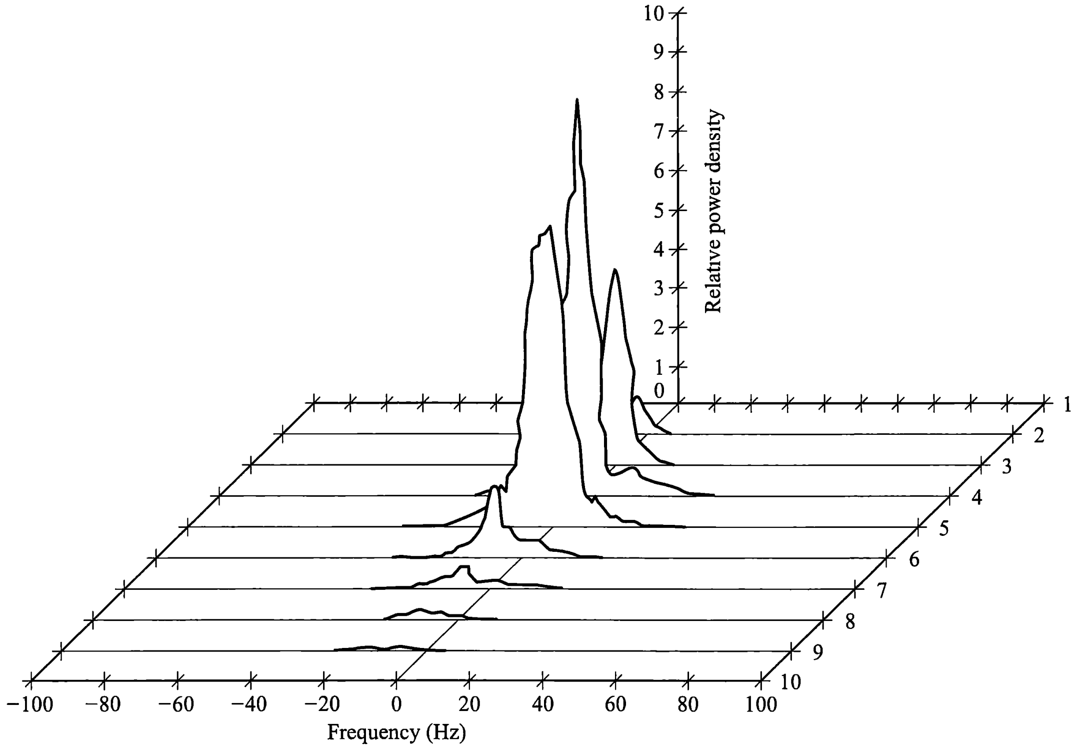


FIGURE 13.1–6
Scattering function of a medium-range tropospheric scatter channel. The taps delay increment is 0.1 μs.

where $J_0(\cdot)$ is the zero-order Bessel function of the first kind and $f_m = vf_0/c$ is the maximum Doppler frequency, where v is the vehicle speed in meters per second (m/s), f_0 is the carrier frequency, and c is the speed of light (3×10^8 m/s). The Fourier transform of this autocorrelation function yields the Doppler power spectrum. That is

$$\begin{aligned} \mathcal{S}_C(\lambda) &= \int_{-\infty}^{\infty} R_C(\Delta t)e^{-j2\pi\lambda\,\Delta t}\,d\Delta t \\ &= \int_{-\infty}^{\infty} J_0(2\pi f_m\,\Delta t)e^{-j2\pi\lambda\,\Delta t}\,d\Delta t \\ &= \begin{cases} \dfrac{1}{\pi f_m}\dfrac{1}{\sqrt{1-(f/f_m)^2}} & |f| \le f_m \\ 0 & |f| > f_m \end{cases} \end{aligned}$$

The graph of $\mathcal{S}_C(\lambda)$ is shown in Figure 13.1–8.

13.1–2 Statistical Models for Fading Channels

There are several probability distributions that can be considered in attempting to model the statistical characteristics of the fading channel. When there are a large number of scatterers in the channel that contribute to the signal at the receiver, as is the case in

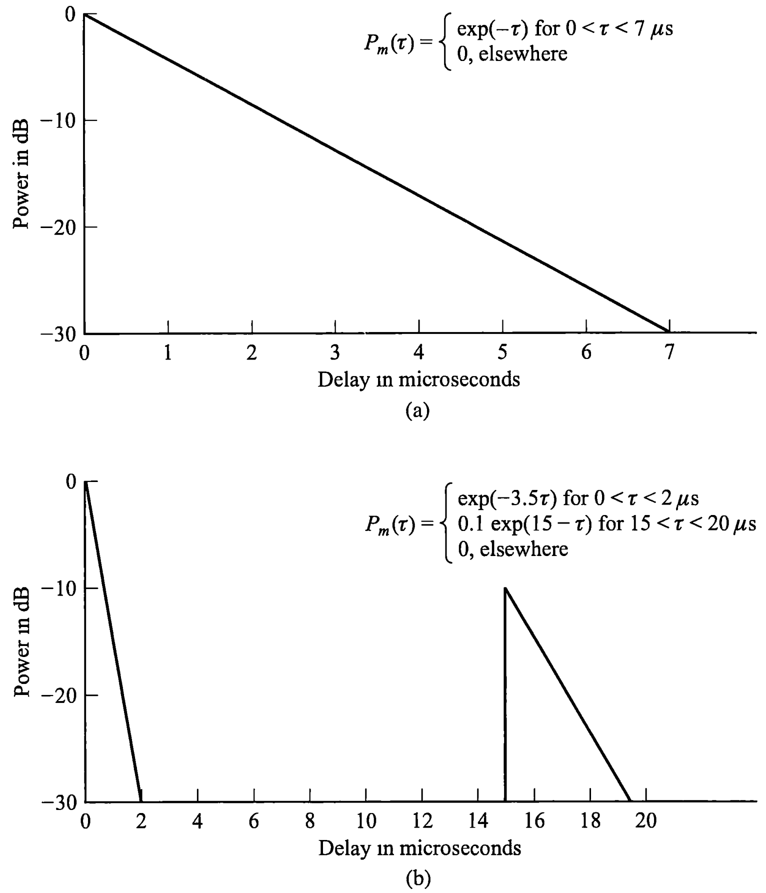


FIGURE 13.1–7

Cost 207 average power delay profiles: (a) typical delay profile for suburban and urban areas; (b) typical "bad"-case delay profile for hilly terrain. [*From Cost 207 Document 207 TD (86)51 rev 3.*]

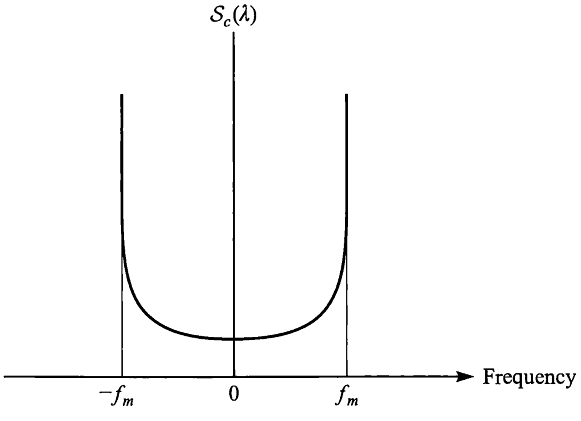


FIGURE 13.1–8

Model of Doppler spectrum for a mobile radio channel.

ionospheric or tropospheric signal propagation, application of the central limit theorem leads to a Gaussian process model for the channel impulse response. If the process is zero-mean, then the envelope of the channel response at any time instant has a Rayleigh probability distribution and the phase is uniformly distributed in the interval $(0, 2\pi)$.

That is

$$p_R(r) = \frac{2r}{\Omega} e^{-r^2/\Omega}, \qquad r \geq 0 \tag{13.1–23}$$

where

$$\Omega = E(R^2) \tag{13.1–24}$$

We observe that the Rayleigh distribution is characterized by the single parameter $E(R^2)$.

An alternative statistical model for the envelope of the channel response is the Nakagami-m distribution given by the PDF in Equation 2.3–67. In contrast to the Rayleigh distribution, which has a single parameter that can be used to match the fading channel statistics, the Nakagami-m is a two-parameter distribution, involving the parameter m and the second moment $\Omega = E(R^2)$. As a consequence, this distribution provides more flexibility and accuracy in matching the observed signal statistics. The Nakagami-m distribution can be used to model fading channel conditions that are either more or less severe than the Rayleigh distribution, and it includes the Rayleigh distribution as a special case ($m = 1$). For example, Turin et al. (1972) and Suzuki (1977) have shown that the Nakagami-m distribution provides the best fit for data signals received in urban radio multipath channels.

The Rice distribution is also a two-parameter distribution. It may be expressed by the PDF given in Equation 2.3–56, where the parameters are s and σ^2, where s^2 is called the *noncentrality parameter* in the equivalent chi-square distribution. It represents the power in the nonfading signal components, sometimes called *specular components*, of the received signal.

There are many radio channels in which fading is encountered that are basically line-of-sight (LOS) communication links with multipath components arising from secondary reflections, or signal paths, from surrounding terrain. In such channels, the number of multipath components is small, and, hence, the channel may be modeled in a somewhat simpler form. We cite two channel models as examples.

As the first example, let us consider an airplane to ground communication link in which there is the direct path and a single multipath component at a delay t_0 relative to the direct path. The impulse response of such a channel may be modeled as

$$c(\tau; t) = \alpha\delta(\tau) + \beta(t)\delta[\tau - \tau_0(t)] \tag{13.1–25}$$

where α is the attenuation factor.of the direct path and $\beta(t)$ represents the time-variant multipath signal component resulting from terrain reflections. Often, $\beta(t)$ can be characterized as a zero-mean Gaussian random process. The transfer function for this channel model may be expressed as

$$C(f; t) = \alpha + \beta(t)e^{-j2\pi f \tau_0(t)} \tag{13.1–26}$$

This channel fits the Ricean fading model defined previously. The direct path with attenuation α represents the specular component and $\beta(t)$ represents the Rayleigh fading component.

A similar model has been found to hold for microwave LOS radio channels used for long-distance voice and video transmission by telephone companies throughout the world. For such channels, Rummler (1979) has developed a three-path model based on channel measurements performed on typical LOS links in the 6-GHz frequency band. The differential delay on the two multipath components is relatively small, and, hence, the model developed by Rummler is one that has a channel transfer function

$$C(f) = \alpha[1 - \beta e^{-j2\pi(f-f_0)\tau_0}] \qquad (13.1\text{–}27)$$

where α is the overall attenuation parameter, β is called a shape parameter which is due to the multipath components, f_0 is the frequency of the fade minimum, and τ_0 is the relative time delay between the direct and the multipath components. This simplified model was used to fit data derived from channel measurements.

Rummler found that the parameters α and β may be characterized as random variables that, for practical purposes, are nearly statistically independent. From the channel measurements, he found that the distribution of β has the form $(1 - \beta)^{2.3}$. The distribution of α is well modeled by the lognormal distribution, i.e., $-\log \alpha$ is Gaussian. For $\beta > 0.5$, the mean of $-20 \log \alpha$ was found to be 25 dB and the standard deviation was 5 dB. For smaller values of β, the mean decreases to 15 dB. The delay parameter determined from the measurements was $\tau_0 = 6.3$ ns. The magnitude-square response of $C(f)$ is

$$|C(f)|^2 = \alpha^2[1 + \beta^2 - 2\beta \cos 2\pi(f - f_0)\tau_0] \qquad (13.1\text{–}28)$$

$|C(f)|$ is plotted in Figure 13.1–9 as a function of the frequency $f - f_0$ for $\tau_0 = 6.3$ ns. Note that the effect of the multipath component is to create a deep attenuation at $f = f_0$ and at multiples of $1/\tau_0 \approx 159$ MHz. By comparison, the typical channel bandwidth is 30 MHz. This model was used by Lundgren and Rummler (1979) to determine the error rate performance of digital radio systems.

Propagation models for mobile radio channels In the link budget calculations that were described in Section 4.10–2, we had characterized the path loss of radio waves propagating through free space as being inversely proportional to d^2, where d is the distance between the transmitter and the receiver. However, in a mobile radio

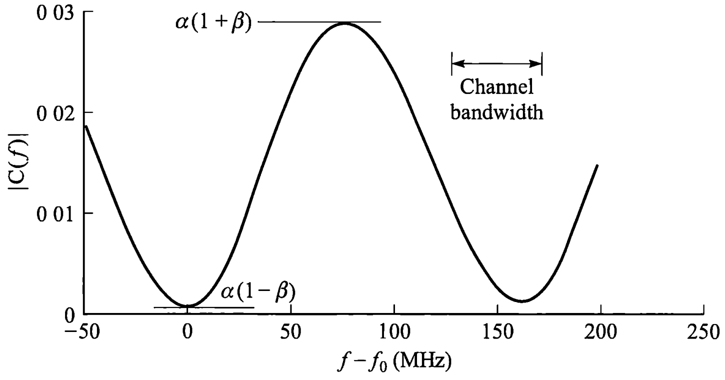


FIGURE 13.1–9
Magnitude frequency response of LOS channel model.

channel, propagation is generally neither free space nor line of sight. The mean path loss encountered in mobile radio channels may be characterized as being inversely proportional to d^p, where $2 \leq p \leq 4$, with d^4 being a worst-case model. Consequently, the path loss is usually much more severe compared to that of free space.

There are a number of factors affecting the path loss in mobile radio communications. Among these factors are base station antenna height, mobile antenna height, operating frequency, atmospheric conditions, and presence or absence of buildings and trees. Various mean path loss models have been developed that incorporate such factors. For example, a model for a large city in an urban area is the Hata model, in which the mean path loss is expressed as

$$\begin{aligned}\text{Loss in dB} &= 69.55 + 26.16\log_{10} f - 13.82\log_{10} h_t - a(h_r) \\ &\quad + (44.9 - 6.55\log_{10} h_t)\log_{10} d\end{aligned} \tag{13.1–29}$$

where f is the operating frequency in MHz ($150 < f < 1500$), h_t is the transmitter antenna height in meters ($30 < h_t < 200$), h_r is the receiver antenna height in meters ($1 < h_r < 10$), d is the distance between transmitter and receiver in km ($1 < d < 20$), and

$$a(h_r) = 3.2(\log_{10} 11.75h_r)^2 - 4.97, \qquad f \geq 400\,\text{MHz} \tag{13.1–30}$$

Another problem with mobile radio propagation is the effect of shadowing of the signal due to large obstructions, such as large buildings, trees, and hilly terrain between the transmitter and the receiver. Shadowing is usually modeled as a multiplicative and, generally, slowly time varying random process. That is, the received signal may be characterized mathematically as

$$r(t) = A_0 g(t) s(t) \tag{13.1–31}$$

where A_0 represents the mean path loss, $s(t)$ is the transmitted signal, and $g(t)$ is a random process that represents the shadowing effect. At any time instant, the shadowing process is modeled statistically as lognormally distributed. The probability density function for the lognormal distribution is

$$p(g) = \begin{cases} \dfrac{1}{\sqrt{2\pi\sigma^2}.g} e^{-(\ln g - \mu)^2/2\sigma^2} & (g \geq 0) \\ 0 & (g < 0) \end{cases} \tag{13.1–32}$$

If we define a new random variable X as $X = \ln g$, then

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2}, \qquad -\infty < x < \infty \tag{13.1–33}$$

The random variable X represents the path loss measured in dB, μ is the mean path loss in dB, and σ is the standard deviation of the path loss in dB. For typical cellular and microcellular environments, σ is in the range of 5–12 dB.

13.2

THE EFFECT OF SIGNAL CHARACTERISTICS ON THE CHOICE OF A CHANNEL MODEL

Having discussed the statistical characterization of time-variant multipath channels generally in terms of the correlation functions describe in Section 13.1, we now consider the effect of signal characteristics on the selection of a channel model that is appropriate for the specified signal. Thus, let $s_l(t)$ be the equivalent lowpass signal transmitted over the channel and let $S_l(f)$ denote its frequency content. Then the equivalent lowpass received signal, exclusive of additive noise, may be expressed either in terms of the time-domain variables $c(\tau; t)$ and $s_l(t)$ as

$$r_l(t) = \int_{-\infty}^{\infty} c(\tau; t) s_l(t - \tau)\, d\tau \tag{13.2–1}$$

or in terms of the frequency functions $C(f; t)$ and $S_l(f)$ as

$$r_l(t) = \int_{-\infty}^{\infty} C(f; t) S_l(f) e^{j2\pi f t}\, df \tag{13.2–2}$$

Suppose we are transmitting digital information over the channel by modulating (either in amplitude, or in phase, or both) the basic pulse $s_l(t)$ at a rate $1/T$, where T is the signaling interval. It is apparent from Equation 13.2–2 that the time-variant channel characterized by the transfer function $C(f; t)$ distorts the signal $S_l(f)$. If $S_l(f)$ has a bandwidth W greater than the coherence bandwidth $(\Delta f)_c$ of the channel, $S_l(f)$ is subjected to different gains and phase shifts across the band. In such a case, the channel is said to be *frequency-selective*. Additional distortion is caused by the time variations in $C(f; t)$. This type of distortion is evidenced as a variation in the received signal strength, and has been termed *fading*. It should be emphasized that the frequency selectivity and fading are viewed as two different types of distortion. The former depends on the multipath spread or, equivalently, on the coherence bandwidth of the channel relative to the transmitted signal bandwidth W. The latter depends on the time variations of the channel, which are grossly characterized by the coherence time $(\Delta t)_c$ or, equivalently, by the Doppler spread B_d.

The effect of the channel on the transmitted signal $s_l(t)$ is a function of our choice of signal bandwidth and signal duration. For example, if we select the signaling interval T to satisfy the condition $T \gg T_m$, the channel introduces a negligible amount of intersymbol interference. If the bandwidth of the signal pulse $s_l(t)$ is $W \approx 1/T$, the condition $T \gg T_m$ implies that

$$W \ll \frac{1}{T_m} \approx (\Delta f)_c \tag{13.2–3}$$

That is, the signal bandwidth W is much smaller than the coherence bandwidth of the channel. Hence, the channel is *frequency-nonselective*. In other words, all the frequency components in $S_l(f)$ undergo the same attenuation and phase shift in transmission through the channel. But this implies that, within the bandwidth occupied by $S_l(f)$, the time-variant transfer function $C(f; t)$ of the channel is a complex-valued constant in the frequency variable. Since $S_l(f)$ has its frequency content concentrated in the vicinity of $f = 0$, $C(f; t) = C(0; t)$. Consequently, Equation 13.2–2 reduces to

$$\begin{aligned} r_l(t) &= C(0; t) \int_{-\infty}^{\infty} S_l(f) e^{j2\pi f t}\, df \\ &= C(0; t) s_l(t) \end{aligned} \tag{13.2–4}$$

Thus, when the signal bandwidth W is much smaller than the coherence bandwidth $(\Delta f)_c$ of the channel, the received signal is simply the transmitted signal multiplied by a complex-valued random process $C(0; t)$, which represents the time-variant characteristics of the channel. In this case, we say that the multipath components in the received are not resolvable because $W \ll (\Delta f)_c$.

The transfer function $C(0; t)$ for a frequency-nonselective channel may be expressed in the form

$$C(0; t) = \alpha(t) e^{j\phi(t)} \tag{13.2–5}$$

where $\alpha(t)$ represents the envelope and $\phi(t)$ represents the phase of the equivalent lowpass channel. When $C(0; t)$ is modeled as a zero-mean complex-valued Gaussian random process, the envelope $\alpha(t)$ is Rayleigh-distributed for any fixed value of t and $\phi(t)$ is uniformly distributed over the interval $(-\pi, \pi)$. The rapidity of the fading on the frequency-nonselective channel is determined either from the correlation function $R_C(\Delta t)$ or from the Doppler power spectrum $\mathcal{S}_C(\lambda)$. Alternatively, either of the channel parameters $(\Delta t)_c$ or B_d can be used to characterize the rapidity of the fading.

For example, suppose it is possible to select the signal bandwidth W to satisfy the condition $W \ll (\Delta f)_c$ and the signaling interval T to satisfy the condition $T \ll (\Delta t)_c$. Since T is smaller than the coherence time of the channel, the channel attenuation and phase shift are essentially fixed for the duration of at least one signaling interval. When this condition holds, we call the channel a *slowly fading channel*. Furthermore, when $W \approx 1/T$, the conditions that the channel be frequency-nonselective and slowly fading imply that the product of T_m and B_d must satisfy the condition $T_m B_d < 1$.

The product $T_m B_d$ is called the *spread factor* of the channel. If $T_m B_d < 1$, the channel is said to be *underspread*; otherwise, it is *overspread*. The multipath spread, the Doppler spread, and the spread factor are listed in Table 13.2–1 for several channels.

TABLE 13.2–1

Multipath Spread, Doppler Spread, and Spread Factor for Several Time-Variant Multipath Channels

Type of channel	Multipath duration, s	Doppler spread, Hz	Spread factor
Shortwave ionospheric propagation (HF)	10^{-3}–10^{-2}	10^{-1}–1	10^{-4}–10^{-2}
Ionospheric propagation under distributed auroral conditions (HF)	10^{-3}–10^{-2}	10–100	10^{-2}–1
Ionospheric forward scatter (VHF)	10^{-4}	10	10^{-3}
Tropospheric scatter (SHF)	10^{-6}	10	10^{-5}
Orbital scatter (X band)	10^{-4}	10^{3}	10^{-1}
Moon at max. libration ($f_0 = 0.4$ kmc)	10^{-2}	10	10^{-1}

We observe from this table that several radio channels, including the moon when used as a passive reflector, are underspread. Consequently, it is possible to select the signal $s_l(t)$ such that these channels are frequency-nonselective and slowly fading. The slow-fading condition implies that the channel characteristics vary sufficiently slowly that they can be measured.

In Section 13.3, we shall determine the error rate performance for binary signaling over a frequency-nonselective slowly fading channel. This channel model is, by far, the simplest to analyze. More importantly, it yields insight into the performance characteristics for digital signaling on a fading channel and serves to suggest the type of signal waveforms that are effective in overcoming the fading caused by the channel.

Since the multipath components in the received signal are not resolvable when the signal bandwidth W is less than the coherence bandwidth $(\Delta f)_c$ of the channel, the received signal appears to arrive at the receiver via a single fading path. On the other hand, we may choose $W \gg (\Delta f)_c$, so that the channel becomes frequency-selective. We shall show later that, under this condition, the multipath components in the received signal are resolvable with a resolution in time delay of $1/W$. Thus, we shall illustrate that the frequency-selective channel can be modeled as a tapped delay line (transversal) filter with time-variant tap coefficients. We shall then derive the performance of binary signaling over such a frequency-selective channel model.

13.3
FREQUENCY-NONSELECTIVE, SLOWLY FADING CHANNEL

In this section, we derive the error rate performance of binary PSK and binary FSK when these signals are transmitted over a frequency-nonselective, slowly fading channel. As described in Section 13.2, the frequency-nonselective channel results in multiplicative distortion of the transmitted signal $s_l(t)$. Furthermore, the condition that the channel fades slowly implies that the multiplicative process may be regarded as a constant during at least one signaling interval. Consequently, if the transmitted signal is $s_l(t)$, the received equivalent lowpass signal in one signaling interval is

$$r_l(t) = \alpha e^{j\phi} s_l(t) + z(t), \qquad 0 \le t \le T \tag{13.3–1}$$

where $z(t)$ represents the complex-valued white Gaussian noise process corrupting the signal.

Let us assume that the channel fading is sufficiently slow that the phase shift ϕ can be estimated from the received signal without error. In that case, we can achieve ideal coherent detection of the received signal. Thus, the received signal can be processed by passing it through a matched filter in the case of binary PSK or through a pair of matched filters in the case of binary FSK. One method that we can use to determine the performance of the binary communication systems is to evaluate the decision variables and from these determine the probability of error. However, we have already done this for a fixed (time-invariant) channel. That is, for a fixed attenuation α, we know the probability of error for binary PSK and binary FSK. From Equation 4.3–13, the expression for the error rate of binary PSK as a function of the received SNR γ_b is

$$P_b(\gamma_b) = Q\left(\sqrt{2\gamma_b}\right) \tag{13.3–2}$$

where $\gamma_b = \alpha^2 \mathcal{E}_b / N_0$. The expression for the error rate of binary FSK, detected coherently, is given by Equation 4.2–32 as

$$P_b(\gamma_b) = Q\left(\sqrt{\gamma_b}\right) \tag{13.3–3}$$

We view Equations 13.3–2 and 13.3–3 as conditional error probabilities, where the condition is that α is fixed. To obtain the error probabilities when α is random, we must average $P_b(\gamma_b)$, given in Equations 13.3–2 and 13.3–3, over the probability density function of γ_b. That is, we must evaluate the integral

$$P_b = \int_0^\infty P_b(\gamma_b) p(\gamma_b)\, d\gamma_b \tag{13.3–4}$$

where $p(\gamma_b)$ is the probability density function of γ_b when α is random.

Rayleigh fading When α is Rayleigh-distributed, α^2 has a chi-square probability distribution with two degrees of freedom. Consequently, γ_b also is chi-square-distributed. It is easily shown that

$$p(\gamma_b) = \frac{1}{\bar{\gamma}_b} e^{-\gamma_b/\bar{\gamma}_b}, \qquad \gamma_b \geq 0 \tag{13.3–5}$$

where $\overline{\gamma}_b$ is the average signal-to-noise ratio, defined as

$$\overline{\gamma}_b = \frac{\mathcal{E}_b}{N_0} E(\alpha^2) \tag{13.3–6}$$

The term $E(\alpha^2)$ is simply the average value of α^2.

Now we can substitute Equation 13.3–5 into Equation 13.3–4 and carry out the integration for $P_b(\gamma_b)$ as given by Equations 13.3–2 and 13.3–3. The result of this integration for binary PSK is (see Problems 4.44 and 4.50)

$$P_b = \frac{1}{2}\left(1 - \sqrt{\frac{\overline{\gamma}_b}{1 + \bar{\gamma}_b}}\right) \tag{13.3–7}$$

If we repeat the integration with $P_b(\gamma_b)$ given by Equation 13.3–3, we obtain the probability of error for binary FSK, detected coherently, in the form

$$P_b = \frac{1}{2}\left(1 - \sqrt{\frac{\overline{\gamma}_b}{2 + \bar{\gamma}_b}}\right) \tag{13.3–8}$$

In arriving at the error rate results in Equations 13.3–7 and 13.3–8, we have assumed that the estimate of the channel phase shift, obtained in the presence of slow fading, is noiseless. Such an ideal condition may not hold in practice. In such a case, the expressions in Equations 13.3–7 and 13.3–8 should be viewed as representing the best achievable performance in the presence of Rayleigh fading. In Appendix C we consider the problem of estimating the phase in the presence of noise and we evaluate the error rate performance of binary and multiphase PSK.

On channels for which the fading is sufficiently rapid to preclude the estimation of a stable phase reference by averaging the received signal phase over many signaling intervals, DPSK, is an alternative signaling method. Since DPSK requires phase stability over only two consecutive signaling intervals, this modulation technique is quite robust in the presence of signal fading. In deriving the performance of binary DPSK for a fading channel, we begin again with the error probability for a nonfading channel, which is

$$P_b(\gamma_b) = \tfrac{1}{2}e^{-\gamma_b} \tag{13.3–9}$$

This expression is substituted into the integral in Equation 13.3–4 along with $p(\gamma_b)$ obtained from Equation 13.3–5. Evaluation of the resulting integral yields the probability of error for binary DPSK, in the form

$$P_b = \frac{1}{2(1+\overline{\gamma}_b)} \tag{13.3–10}$$

If we choose not to estimate the channel phase shift at all, but instead employ a noncoherent (envelope or square-law) detector with binary, orthogonal FSK signals, the error probability for a nonfading channel is

$$P_b(\gamma_b) = \tfrac{1}{2}e^{-\gamma_b/2} \tag{13.3–11}$$

When we average $P_b(\gamma_b)$ over the Rayleigh fading channel attenuation, the resulting error probability is

$$P_b = \frac{1}{2+\overline{\gamma}_b} \tag{13.3–12}$$

The error probabilities in Equations 13.3–7, 13.3–8, 13.3–10, and 13.3–12 are illustrated in Figure 13.3–1. In comparing the performance of the four binary signaling systems, we focus our attention on the probabilities of error for large SNR, i.e., $\overline{\gamma}_b \gg 1$. Under this condition, the error rates in Equations 13.3–7, 13.3–8, 13.3–10, and 13.3–12 simplify to

$$P_b \approx \begin{cases} 1/4\overline{\gamma}_b & \text{for coherent PSK} \\ 1/2\overline{\gamma}_b & \text{for coherent, orthogonal FSK} \\ 1/2\overline{\gamma}_b & \text{for DPSK} \\ 1/\overline{\gamma}_b & \text{for noncoherent, orthogonal FSK} \end{cases} \tag{13.3–13}$$

From Equation 13.3–13, we observe that coherent PSK is 3 dB better than DPSK and 6 dB better than noncoherent FSK. More striking, however, is the observtion that the error rates decrease only inversely with SNR. In contrast, the decrease in error rate on a nonfading channel is exponential with SNR. This means that, on a fading channel, the transmitter must transmit a large amount of power in order to obtain a low probability of error. In many cases, a large amount of power is not possible, technically and/or economically. An alternative solution to the problem of obtaining acceptable

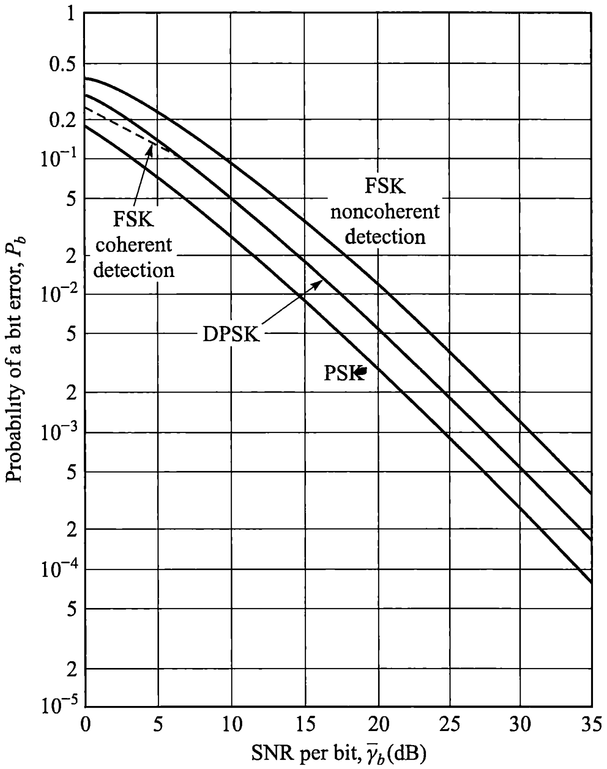


FIGURE 13.3–1
Performance of binary signaling on a Rayleigh fading channel.

performance on a fading channel is the use of redundancy, which can be obtained by means of diversity techniques, as discussed in Section 13.4.

Nakagami fading If α is characterized statistically by the Nakagami-m distribution, the random variable $\gamma = \alpha^2 \mathcal{E}_b / N_0$ has the PDF (see Problem 13.14)

$$p(\gamma) = \frac{m^m}{\Gamma(m)\bar{\gamma}^m} \gamma^{m-1} e^{-m\gamma/\bar{\gamma}} \tag{13.3–14}$$

where $\bar{\gamma} = E(\alpha^2)\mathcal{E}/N_0$.

The average probability of error for any of the modulation methods is simply obtained by averaging the appropriate error probability for a nonfading channel over the fading signal statistics.

As an example of the performance obtained with Nakagami-m fading statistics, Figure 13.3–2 illustrates the probability of error of binary PSK with m as a parameter. We recall that $m = 1$ corresponds to Rayleigh fading. We observe that the performance improves as m is increased above $m = 1$, which is indicative of the fact that the fading is less severe. On the other hand, when $m < 1$, the performance is worse than Rayleigh fading.

Other fading signal statistics Following the procedure describe above, one can determine the performance of the various modulation methods for other types of fading signal statistics, such as Ricean Fading.

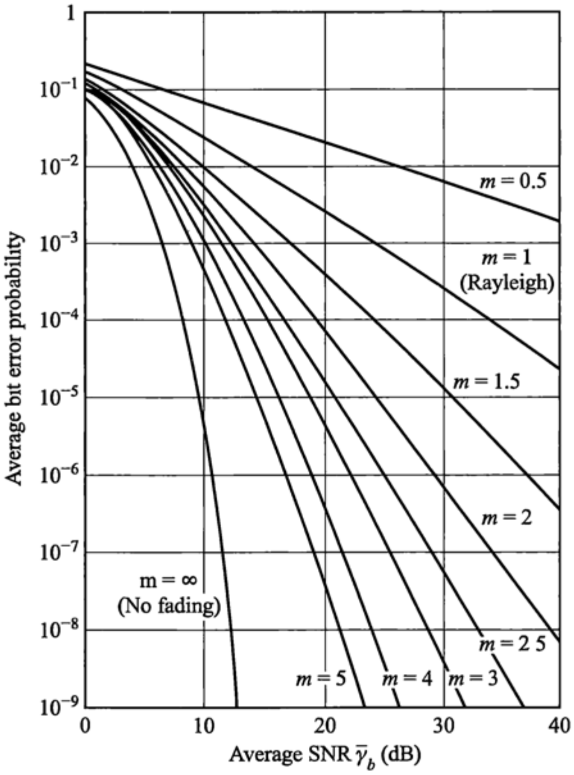


FIGURE 13.3–2
Average error probability for two-phase PSK with Nakagami fading.

Error probability results for Rice-distributed fading statistics can be found in the paper by Lindsey (1964), while for Nakagami-*m* fading statistics, the reader may refer to the papers by Esposito (1967), Miyagaki et al. (1978), Charash (1979), Al-Hussaini et al. (1985), and Beaulieu and Abu-Dayya (1991).

13.4

DIVERSITY TECHNIQUES FOR FADING MULTIPATH CHANNELS

Diversity techniques are based on the notion that errors occur in reception when the channel attenuation is large, i.e., when the channel is in a deep fade. If we can supply to the receiver several replicas of the same information signal transmitted over independently fading channels, the probability that all the signal components will fade simultaneously is reduced considerably. That is, if p is the probability that any one signal will fade below some critical value, then p^L is the probability that all L independently fading replicas of the same signal will fade below the critical value. There are several ways in which we can provide the receiver with L independently fading replicas of the same information-bearing signal.

One method is to employ *frequency diversity*. That is, the same information-bearing signal is transmitted on L carriers, where the separation between successive carriers equals or exceeds the coherence bandwidth $(\Delta f)_c$ of the channel.

A second method for achieving L independently fading versions of the same information-bearing signal is to transmit the signal in L different time slots, where the separation between successive time slots equals or exceeds the coherence time $(\Delta t)_c$ of the channel. This method is called *time diversity*.

Note that the fading channel fits the model of a bursty error channel. Furthermore, we may view the transmission of the same information either at different frequencies or in difference time slots (or both) as a simple form of repetition coding. The separation of the diversity transmissions in time by $(\Delta t)_c$ or in frequency by $(\Delta f)_c$ is basically a form of block-interleaving the bits in the repetition code in an attempt to break up the error bursts and, thus, to obtain independent errors. Later in the chapter, we shall demonstrate that, in general, repetition coding is wasteful of bandwidth when compared with nontrivial coding.

Another commonly used method for achieving diversity employs multiple antennas. For example, we may employ a single transmitting antenna and multiple receiving antennas. The latter must be spaced sufficiently far apart that the multipath components in the signal have significantly different propagation delays at the antennas. Usually a separation of a few wavelengths is required between two antennas in order to obtain signals that fade independently.

A more sophisticated method for obtaining diversity is based on the use of a signal having a bandwidth much greater than the coherence bandwidth $(\Delta f)_c$ of the channel. Such a signal with bandwidth W will resolve the multipath components and, thus, provide the receiver with several independently fading signal paths. The time resolution is $1/W$. Consequently, with a multipath spread of T_m seconds, there are $T_m W$ resolvable signal components. Since $T_m \approx 1/(\Delta f)_c$, the number of resolvable signal components may also be expressed as $W/(\Delta f)_c$. Thus, the use of a wideband signal may be viewed as just another method for obtaining frequency diversity of order $L \approx W/(\Delta f)_c$. The optimum demodulator for processing the wideband signal will be derived in Section 13.5. It is called a *RAKE correlator* or a *RAKE matched filter* and was invented by Price and Green (1958).

There are other diversity techniques that have received some consideration in practice, such as angle-of-arrival diversity and polarization diversity. However, these have not been as widely used as those described above.

13.4–1 Binary Signals

We shall now determine the error rate performance for a binary digital communication system with diversity. We begin by describing the mathematical model for the communication system with diversity. First of all, we assume that there are L diversity channels, carrying the same information-bearing signal. Each channel is assumed to be frequency-nonselective and slowly fading with Rayleigh-distributed envelope statistics. The fading processes among the L diversity channels are assumed to be mutually statistically independent. The signal in each channel is corrupted by an additive zero-mean white Gaussian noise process. The noise processes in the L channels are assumed to be mutually statistically independent, with identical autocorrelation functions. Thus, the equivalent low-pass received signals for the L channels can be expressed in the form

$$r_{lk}(t) = \alpha_k e^{j\phi_k} s_{km}(t) + z_k(t), \qquad k = 1, 2, \ldots, L, \qquad m = 1, 2 \qquad (13.4\text{–}1)$$

where $\{\alpha_k e^{j\phi_k}\}$ represent the attenuation factors and phase shifts for the L channels, $s_{km}(t)$ denotes the mth signal transmitted on the kth channel, and $z_k(t)$ denotes the additive white Gaussian noise on the kth channel. All signals in the set $\{s_{km}(t)\}$ have the same energy.

The optimum demodulator for the signal received from the kth channel consists of two matched filters, one having the impulse response

$$b_{k1}(t) = s_{k1}^*(T - t) \tag{13.4–2}$$

and the other having the impulse response

$$b_{k2}(t) = s_{k2}^*(T - t) \tag{13.4–3}$$

Of course, if binary PSK is the modulation method used to transmit the information, then $s_{k1}(t) = -s_{k2}(t)$. Consequently, only a single matched filter is required for binary PSK. Following the matched filters is a combiner that forms the two decision variables. The combiner that achieves the best performance is one in which each matched filter output is multiplied by the corresponding complex-valued (conjugate) channel gain $\alpha_k e^{-j\phi_k}$. The effect of this multiplication is to compensate for the phase shift in the channel and to weight the signal by a factor that is proportional to the signal strength. Thus, a strong signal carries a larger weight than a weak signal. After the complex-valued weighting operation is performed, two sums are formed. One consists of the real parts of the weighted outputs from the matched filters corresponding to a transmitted 0. The second consists of the real part of the outputs from the matched filters corresponding to a transmitted 1. This optimum combiner is called a *maximal ratio combiner* by Brennan (1959). Of course, the realization of this optimum combiner is based on the assumption that the channel attenuations $\{\alpha_k\}$ and the phase shifts $\{\phi_k\}$ are known perfectly. That is, the estimates of the parameters $\{\alpha_k\}$ and $\{\phi_k\}$ contain no noise. (The effect of noisy estimates on the error rate performance of multiphase PSK is considered in Appendix C.)

A block diagram illustrating the model for the binary digital communication system described above is shown in Figure 13.4–1.

Let us first consider the performance of binary PSK with Lth-order diversity. The output of the maximal ratio combiner can be expressed as a single decision variable in the form

$$\begin{aligned} U &= \operatorname{Re}\left(2\mathcal{E}\sum_{k=1}^{L}\alpha_k^2 + \sum_{k=1}^{L}\alpha_k N_k\right) \\ &= 2\mathcal{E}\sum_{k=1}^{L}\alpha_k^2 + \sum_{k=1}^{L}\alpha_k N_{kr} \end{aligned} \tag{13.4–4}$$

where N_{kr} denotes the real part of the complex-valued Gaussian noise variable

$$N_k = e^{-j\phi_k}\int_0^T z_k(t)s_k^*(t)\,dt \tag{13.4–5}$$

We follow the approach used in Section 13.3 in deriving the probability of error. That is, the probability of error conditioned on a fixed set of attenuation factors $\{\alpha_k\}$ is obtained

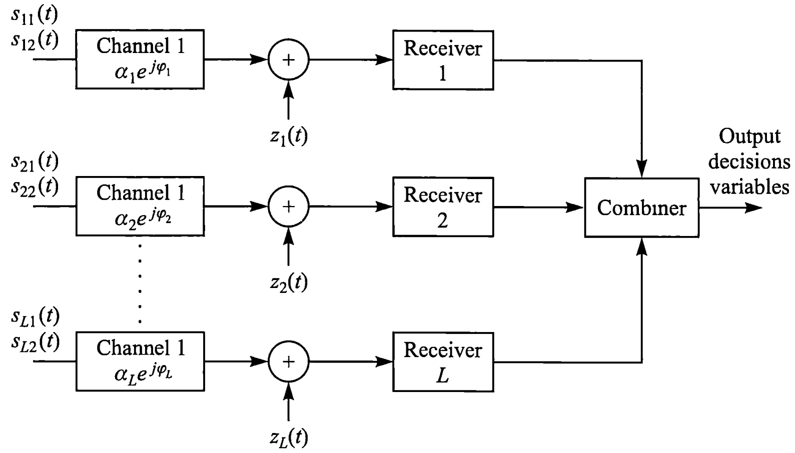

$s_{11}(t)$
$s_{12}(t)$
Channel 1
$\alpha_1 e^{j\varphi_1}$
$z_1(t)$
Receiver 1
$s_{21}(t)$
$s_{22}(t)$
Channel 1
$\alpha_2 e^{j\varphi_2}$
$z_2(t)$
Receiver 2
$s_{L1}(t)$
$s_{L2}(t)$
Channel 1
$\alpha_L e^{j\varphi_L}$
$z_L(t)$
Receiver L
Combıner
Output decisions variables

FIGURE 13.4–1
Model of binary digital communication system with diversity.

first. Then the conditional probability of error is averaged over the probability density function of the $\{\alpha_k\}$.

Rayleigh fading For a fixed set of $\{\alpha_k\}$ the decision variable U is Gaussian with mean

$$E(U) = 2\mathcal{E} \sum_{k=1}^{L} \alpha_k^2 \tag{13.4–6}$$

and variance

$$\sigma_U^2 = 2\mathcal{E} N_0 \sum_{k=1}^{L} \alpha_k^2 \tag{13.4–7}$$

For these values of the mean and variance, the probability that U is less than zero is simply

$$P_b(\gamma_b) = Q\left(\sqrt{2\gamma_b}\right) \tag{13.4–8}$$

where the SNR per bit, γ_b, is given as

$$\begin{aligned} \gamma_b &= \frac{\mathcal{E}}{N_0} \sum_{k=1}^{L} \alpha_k^2 \\ &= \sum_{k=1}^{L} \gamma_k \end{aligned} \tag{13.4–9}$$

where $\gamma_k = \mathcal{E}\alpha_k^2/N_0$ is the instantaneous SNR on the kth channel. Now we must determine the probability density function $p(\gamma_b)$. This function is most easily determined via the characteristic function of γ_b. First of all, we note that for $L = 1$, $\gamma_b \equiv \gamma_1$ has a chi-square probability density function given in Equation 13.3–5. The characteristic function of γ_1 is easily shown to be

$$\Phi_{\gamma_1}(v) = E(e^{jv\gamma_1}) = \frac{1}{1 - jv\overline{\gamma}_c} \tag{13.4–10}$$

where $\overline{\gamma}_c$ is the average SNR per channel, which is assumed to be identical for all channels. That is,

$$\overline{\gamma}_c = \frac{\mathcal{E}}{N_0} E(\alpha_k^2) \tag{13.4–11}$$

independent of k. This assumption applies for the results throughout this section. Since the fading on the L channels is mutually statistically independent, the $\{\gamma_k\}$ are statistically independent, and, hence, the characteristic function for the sum γ_b is simply the result in Equation 13.4–10 raised to the Lth power, i.e.,

$$\Phi_{\gamma_b}(v) = \frac{1}{(1 - jv\overline{\gamma}_c)^L} \tag{13.4–12}$$

But this is the characteristic function of a chi-square-distributed random variable with $2L$ degrees of freedom. It follows from Equation 2.3–21 that the probability density function $p(\gamma_b)$ is

$$p(\gamma_b) = \frac{1}{(L-1)!\overline{\gamma}_c^L} \gamma_b^{L-1} e^{-\gamma_b/\overline{\gamma}_c} \tag{13.4–13}$$

The final step in this derivation is to average the conditional error probability given in Equation 13.4–8 over the fading channel statistics. Thus, we evaluate the integral

$$P_b = \int_0^\infty P_2(\gamma_b) p(\gamma_b)\, d\gamma_b \tag{13.4–14}$$

There is a closed-form solution for Equation 13.4–14, which can be expressed as

$$P_b = \left[\tfrac{1}{2}(1-\mu)\right]^L \sum_{k=0}^{L-1} \binom{L-1+k}{k} \left[\tfrac{1}{2}(1+\mu)\right]^k \tag{13.4–15}$$

where, by definition

$$\mu = \sqrt{\frac{\overline{\gamma}_c}{1+\overline{\gamma}_c}} \tag{13.4–16}$$

When the average SNR per channel, $\overline{\gamma}_c$, satisfies the condition $\overline{\gamma}_c \gg 1$, the term $\frac{1}{2}(1+\mu) \approx 1$ and the term $\frac{1}{2}(1-\mu) \approx 1/4\overline{\gamma}_c$. Furthermore,

$$\sum_{k=0}^{L-1} \binom{L-1+k}{k} = \binom{2L-1}{L} \tag{13.4–17}$$

Therefore, when $\overline{\gamma}_c$ is sufficiently large (greater than 10 dB), the probability of error in Equation 13.4–15 can be approximated as

$$P_b \approx \left(\frac{1}{4\overline{\gamma}_c}\right)^L \binom{2L-1}{L} \tag{13.4–18}$$

We observe from Equation 13.4–18 that the probability of error varies as $1/\overline{\gamma}_c$ raised to the Lth power. Thus, with diversity, the error rate decreases inversely with the Lth power of the SNR.

Having obtained the performance of binary PSK with diversity, we now turn our attention to binary, orthogonal FSK that is detected coherently. In this case, the two decision variables at the output of the maximal ratio combiner may be expressed as

$$\begin{aligned} U_1 &= \mathrm{Re}\left(2\mathcal{E}\sum_{k=1}^{L}\alpha_k^2 + \sum_{k=1}^{L}\alpha_k N_{k1}\right) \\ U_2 &= \mathrm{Re}\left(\sum_{k=1}^{L}\alpha_k N_{k2}\right) \end{aligned} \tag{13.4–19}$$

where we have assumed that signal $s_{k1}(t)$ was transmitted and where $\{N_{k1}\}$ and $\{N_{k2}\}$ are the two sets of noise components at the output of the matched filters. The probability of error is simply the probability that $U_2 > U_1$. This computation is similar to the one performed for PSK, except that we now have twice the noise power. Consequently, when the $\{\alpha_k\}$ are fixed, the conditional probability of error is

$$P_b(\gamma_b) = Q\left(\sqrt{\gamma_b}\right) \tag{13.4–20}$$

We use Equation 13.4–13 to average $P_b(\gamma_b)$ over the fading. It is not surprising to find that the result given in Equation 13.4–15 still applies, with $\overline{\gamma}_c$ replaced by $\frac{1}{2}\overline{\gamma}_c$. That is, Equation 13.4–15 is the probability of error for binary, orthogonal FSK with coherent detection, where the parameter μ is defined as

$$\mu = \sqrt{\frac{\overline{\gamma}_c}{2+\overline{\gamma}_c}} \tag{13.4–21}$$

Furthermore, for large values of $\overline{\gamma}_c$, the performance P_b can be approximated as

$$P_b \approx \left(\frac{1}{2\overline{\gamma}_c}\right)^L \binom{2L-1}{L} \tag{13.4–22}$$

In comparing Equation 13.4–22 with Equation 13.4–18, we observe that the 3-dB difference in performance between PSK and orthogonal FSK with coherent detection, which exists in a nonfading, nondispersive channel, is the same also in a fading channel.

In the above discussion of binary PSK and FSK, detected coherently, we assumed that noiseless estimates of the complex-valued channel parameters $\{\alpha_k e^{j\phi_k}\}$ were used at the receiver. Since the channel is time-variant, the parameters $\{\alpha_k e^{j\phi_k}\}$ cannot be estimated perfectly. In fact, on some channels, the time variations may be sufficiently fast to preclude the implementation of coherent detection. In such a case, we should consider using either DPSK or FSK with noncoherent detection.

Let us consider DPSK first. In order for DPSK to be a viable digital signaling method, the channel variations must be sufficiently slow so that the channel phase shifts $\{\phi_k\}$ do not change appreciably over two consecutive signaling intervals. In our analysis, we assume that the channel parameters $\{\alpha_k e^{j\phi_k}\}$ remain constant over two successive signaling intervals. Thus the combiner for binary DPSK will yield as an output the decision variable

$$U = \text{Re}\left[\sum_{k=1}^{L} \left(2\mathcal{E}\alpha_k e^{j\phi_k} + N_{k2}\right)\left(2\mathcal{E}\alpha_k e^{-j\phi_k} + N_{k1}^{*}\right)\right] \tag{13.4–23}$$

where $\{N_{k1}\}$ and $\{N_{k2}\}$ denote the received noise components at the output of the matched filters in the two consecutive signaling intervals. The probability of error is simply the probability that $U < 0$. Since U is a special case of the general quadratic form in complex-valued Gaussian random variables treated in Appendix B, the probability of error can be obtained directly from the results given in that appendix. Alternatively, we may use the error probability given in Equation 11.1–13, which applies to binary DPSK transmitted over L time-invariant channels, and average it over the Rayleigh fading channel statistics. Thus, we have the conditional error probability

$$P_b(\gamma_b) = \left(\tfrac{1}{2}\right)^{2L-1} e^{-\gamma_b} \sum_{k=0}^{L-1} b_k \gamma_b^k \tag{13.4–24}$$

where γ_b is given by Equation 13.4–9 and

$$b_k = \frac{1}{k!} \sum_{n=0}^{L-1-k} \binom{2L-1}{n} \tag{13.4–25}$$

The average of $P_b(\gamma_b)$ over the fading channel statistics given by $p(\gamma_b)$ in Equation 13.4–13 is easily shown to be

$$P_b = \frac{1}{2^{2L-1}(L-1)!(1+\overline{\gamma}_c)^L} \sum_{k=0}^{L-1} b_k (L-1+k)! \left(\frac{\overline{\gamma}_c}{1+\overline{\gamma}_c}\right)^k \tag{13.4–26}$$

We indicate that the result in Equation 13.4–26 can be manipulated into the form given in Equation 13.4–15, which applies also to coherent PSK and FSK. For binary DPSK, the parameter μ in Equation 13.4–15 is defined as (see Appendix C)

$$\mu = \frac{\overline{\gamma}_c}{1+\overline{\gamma}_c} \tag{13.4–27}$$

For $\overline{\gamma}_c \gg 1$, the error probability in Equation 13.4–26 can be approximated by the expression

$$P_b \approx \left(\frac{1}{2\overline{\gamma}_c}\right)^L \binom{2L-1}{L} \tag{13.4–28}$$

Orthogonal FSK with noncoherent detection is the final signaling technique that we consider in this section. It is appropriate for both slow and fast fading. However, the analysis of the performance presented below is based on the assumption that the fading is sufficiently slow so that the channel parameters $\{\alpha_k e^{j\phi_k}\}$ remain constant for the duration of the signaling interval. The combiner for the multichannel signals is a square-law combiner. Its output consists of the two decision variables

$$\begin{aligned} U_1 &= \sum_{k=1}^{L} |2\mathcal{E}\alpha_k e^{j\phi_k} + N_{k1}|^2 \\ U_2 &= \sum_{k=1}^{L} |N_{k2}|^2 \end{aligned} \tag{13.4–29}$$

where U_1 is assumed to contain the signal. Consequently the probability of error is the probability that $U_2 > U_1$.

As in DPSK, we have a choice of two approaches in deriving the performance of FSK with square-law combining. In Section 11.1, we indicated that the expression for the error probability for square-law-combined FSK is the same as that for DPSK with γ_b replaced by $\frac{1}{2}\gamma_b$. That is, the FSK system requires 3 dB of additional SNR to achieve the same performance on a time-invariant channel. Consequently, the conditional error probability for DPSK given in Equation 13.4–24 applies to square-law-combined FSK when γ_b is replaced by $\frac{1}{2}\gamma_b$. Furthermore, the result obtained by averaging Equation 13.4–24 over the fading, which is given by Equation 13.4–26, must also apply to FSK with $\overline{\gamma}_c$ replaced by $\frac{1}{2}\overline{\gamma}_c$. But we also stated previously that Equations 13.4–26 and 13.4–15 are equivalent. Therefore, the error probability given in Equation 13.4–15 also applies to square-law-combined FSK with the parameter μ defined as

$$\mu = \frac{\overline{\gamma}_c}{2 + \overline{\gamma}_c} \tag{13.4–30}$$

An alternative derivation used by Pierce (1958) to obtain the probability that the decision variable $U_2 > U_1$ is just as easy as the method described above. It begins with the probability density functions $p(u_1)$ and $p(u_2)$. Since the complex-valued random variables $\{\alpha_k e^{j\phi_k}\}$, $\{N_{k1}\}$, and $\{N_{k2}\}$ are zero-mean Gaussian-distributed, the decision variables U_1 and U_2 are distributed according to a chi-square probability distribution with $2L$ degrees of freedom. That is,

$$p(u_1) = \frac{1}{(2\sigma_1^2)^L (L-1)!} u_1^{L-1} \exp\left(-\frac{u_1}{2\sigma_1^2}\right) \tag{13.4–31}$$

where

$$\begin{aligned} \sigma_1^2 &= \tfrac{1}{2} E\left(|2\mathcal{E}\alpha_k e^{-j\phi_k} + N_{k1}|^2\right) \\ &= 2\mathcal{E}N_0(1 + \overline{\gamma}_c) \end{aligned}$$

Similarly,

$$p(u_2) = \frac{1}{\left(2\sigma_2^2\right)(L-1)!} u_2^{L-1} \exp\left(-\frac{u_2}{2\sigma_2^2}\right) \tag{13.4–32}$$

where

$$\sigma_2^2 = 2\mathcal{E}N_0$$

The probability of error is just the probability that $U_2 > U_1$. It is left as an exercise for the reader to show that this probability is given by Equation 13.4–15, where μ is defined by Equation 13.4–30.

When $\overline{\gamma}_c \gg 1$, the performance of square-law-detected FSK can be simplified as we have done for the other binary multichannel systems. In this case, the error rate is well approximated by the expression

$$P_b \approx \left(\frac{1}{\overline{\gamma}_c}\right)^L \binom{2L-1}{L} \tag{13.4–33}$$

The error rate performance of PSK, DPSK, and square-law-detected orthogonal FSK is illustrated in Figure 13.4–2 for $L = 1$, 2, and 4. The performance is plotted as a function of the average SNR per bit, $\overline{\gamma}_b$, which is related to the average SNR per channel, $\overline{\gamma}_c$, by the formula

$$\overline{\gamma}_b = L\overline{\gamma}_c \tag{13.4–34}$$

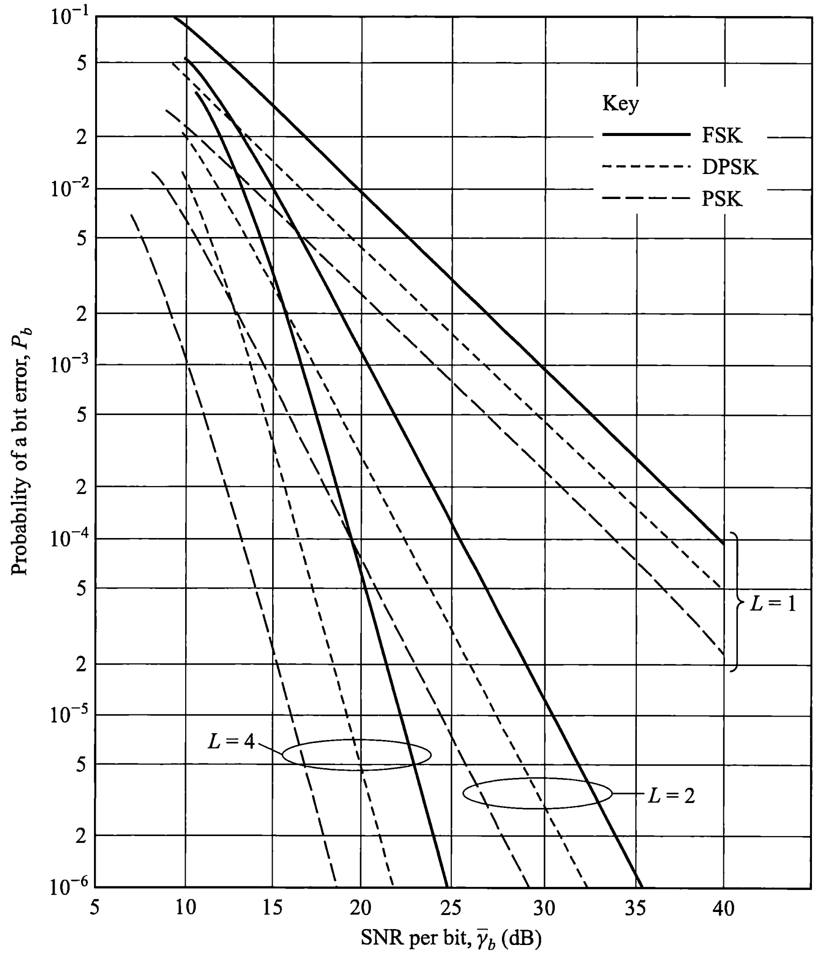


FIGURE 13.4–2
Performance of binary signals with diversity.

The results in Figure 13.4–2 clearly illustrate the advantage of diversity as a means for overcoming the severe penalty in SNR caused by fading.

Nakagami fading It is a simple matter to extend the results of this section to other fading models. We shall briefly consider Nakagami fading. Let us compare the Nakagami PDF for the single-channel SNR parameter $\gamma_b = \alpha^2 \mathcal{E}_b / N_0$, previously given by Equation 13.3–14 as

$$p(\gamma_b) = \frac{1}{\Gamma(m)(\overline{\gamma}_b/m)^m} \gamma_b^{m-1} e^{-\gamma_b/(\overline{\gamma}_b/m)} \tag{13.4–35}$$

with the PDF $p(\gamma_b)$ obtained for the L-channel SNR with Rayleigh fading, given by Equation 13.4–13 as

$$p(\gamma_b) = \frac{1}{(L-1)!\overline{\gamma}_c^L} \gamma_b^{L-1} e^{-\gamma_b/\overline{\gamma}_c} \tag{13.4–36}$$

By noting that $\overline{\gamma}_c = \overline{\gamma}_b/L$ in the case of an Lth order diversity system, it is clear that the two PDFs are identical for $L = m =$ integer. When $L = m = 1$, the two PDFs correspond to a single channel Rayleigh fading system. For the case in which the Nakagami parameter $m = 2$, the performance of the single-channel system is identical to the performance obtained in a Rayleigh fading channel with dual ($L = 2$) diversity. More generally, any single-channel system with Nakagami fading in which the parameter m is an integer, is equivalent to an L-channel diversity system for a Rayleigh fading channel. In view of this equivalence, the characteristic function of a Nakagami-m random variable must be of the form

$$\Phi_{\gamma_b}(v) = \frac{1}{(1 - jv\overline{\gamma}_b/m)^m} \tag{13.4–37}$$

which is consistent with the result given in Equation 13.4–12 for the characteristic function of the combined signal in a system with Lth-order diversity in a Rayleigh fading channel. Consequently, it follows that a K-channel system transmitting in a Nakagami fading channel with independent fading is equivalent to an $L = Km$ channel diversity in a Rayleigh fading channel.

13.4–2 Multiphase Signals

Multiphase signaling over a Rayleigh fading channel is the topic presented in some detail in Appendix C. Our main purpose in this section is to cite the general result for the probability of a symbol error in M-ary PSK and DPSK systems and the probability of a bit error in four-phase PSK and DPSK.

The general result for the probability of a symbol error in M-ary PSK and DPSK is

$$P_e = \frac{(-1)^{L-1}(1-\mu^2)^L}{\pi(L-1)!}\left(\frac{\partial^{L-1}}{\partial b^{L-1}}\left\{\frac{1}{b-\mu^2}\left[\frac{\pi}{M}(M-1)\right.\right.\right.$$
$$\left.\left.\left.-\frac{\mu\sin(\pi/M)}{\sqrt{b-\mu^2\cos^2(\pi/M)}}\cot^{-1}\frac{-\mu\cos(\pi/M)}{\sqrt{b-\mu^2\cos^2(\pi/M)}}\right]\right\}\right)_{b=1} \tag{13.4–38}$$

where

$$\mu = \sqrt{\frac{\overline{\gamma}_c}{1+\overline{\gamma}_c}} \tag{13.4–39}$$

for coherent PSK and

$$\mu = \frac{\overline{\gamma}_c}{1+\overline{\gamma}_c} \tag{13.4–40}$$

for DPSK. Again $\overline{\gamma}_c$ is the average received SNR per channel. The SNR per bit is $\overline{\gamma}_b = L\overline{\gamma}_c/k$, where $k = \log_2 M$.

The bit error rate for four-phase PSK and DPSK is derived on the basis that the pair of information bits is mapped into the four phases according to a Gray code. The expression for the bit error rate derived in Appendix C is

$$P_b = \frac{1}{2}\left[1 - \frac{\mu}{\sqrt{2-\mu^2}}\sum_{k=0}^{L-1}\binom{2k}{k}\left(\frac{1-\mu^2}{4-2\mu^2}\right)^k\right] \tag{13.4–41}$$

where μ is again given by Equations 13.4–39 and 13.4–40 for PSK and DPSK, respectively.

Figure 13.4–3 illustrates the probability of a symbol error of DPSK and coherent PSK for $M = 2, 4$, and 8 with $L = 1$. Note that the difference in performance between DPSK and coherent PSK is approximately 3 dB for all three values of M. In fact, when $\overline{\gamma}_b \gg 1$ and $L = 1$, Equation 13.4–38 is well approximated as

$$P_e \approx \frac{M-1}{(M\log_2 M)[\sin^2(\pi/M)]\overline{\gamma}_b} \tag{13.4–42}$$

for DPSK and as

$$P_e \approx \frac{M-1}{(M\log_2 M)[\sin^2(\pi/M)]2\overline{\gamma}_b} \tag{13.4–43}$$

for PSK. Hence, at high SNR, coherent PSK is 3 dB better than DPSK on a Rayleigh fading channel. This difference also holds as L is increased.

Bit error probabilities are depicted in Figure 13.4–4 for two-phase, four-phase, and eight-phase DPSK signaling with $L = 1$, 2, and 4. The expression for the bit error probability of eight-phase DPSK with Gray encoding is not given here, but it is available in the paper by Proakis (1968). In this case, we observe that the performances for two- and four-phase DPSK are (approximately) the same, while that for eight-phase DPSK is about 3 dB poorer. Although we have not shown the bit error probability for

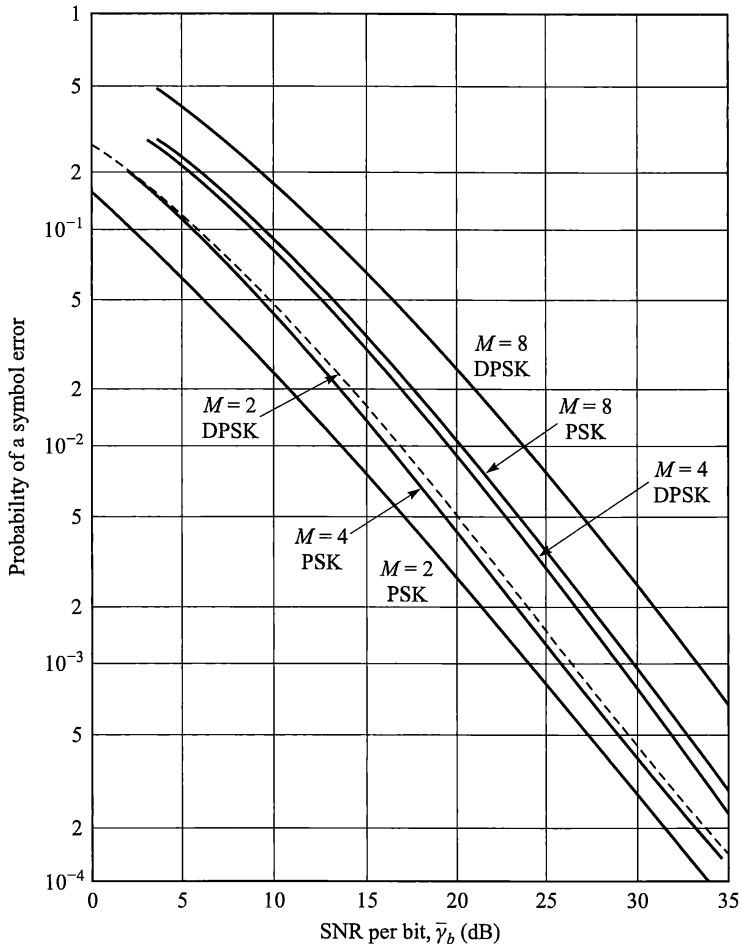


FIGURE 13.4–3
Probability of symbol error for PSK and DPSK for Rayleigh fading.

coherent PSK, it can be demonstrated that two- and four-phase coherent PSK also yield approximately the same performance.

13.4–3 M-ary Orthogonal Signals

In this subsection, we determine the performance of M-ary orthogonal signals transmitted over a Rayleigh fading channel and we assess the advantages of higher-order signal alphabets relative to a binary alphabet. The orthogonal signals may be viewed as M-ary FSK with a minimum frequency separation of an integer multiple of $1/T$, where T is the signaling interval. The same information-bearing signal is transmitted on L diversity channels. Each diversity channel is assumed to be frequency-nonselective and slowly fading, and the fading processes on the L channels are assumed to be mutually statistically independent. An additive white Gaussian noise process corrupts the signal on each diversity channel. We assume that the additive noise processes are mutually statistically independent.

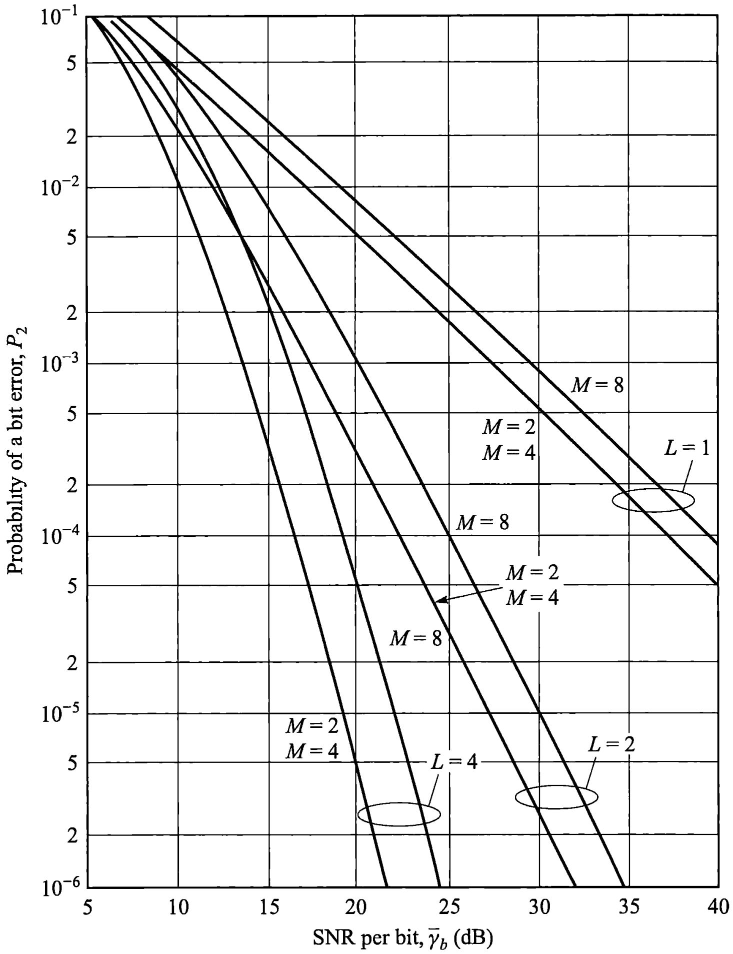


FIGURE 13.4–4
Probability of a bit error for DPSK with diversity for Rayleigh fading.

Although it is relatively easy to formulate the structure and analyze the performance of a maximal ratio combiner for the diversity channels in the M-ary communication system, it is more likely that a practical system would employ noncoherent detection. Consequently, we confine our attention to square-law combining of the diversity signals. The output of the combiner containing the signal is

$$U_1 = \sum_{k=1}^{L} |2\mathcal{E}\alpha_k e^{j\phi_k} + N_{k1}|^2 \tag{13.4–44}$$

while the outputs of the remaining $M - 1$ combiners are

$$U_m = \sum_{k=1}^{L} |N_{km}|^2, \qquad m = 2, 3, 4, \ldots, M \tag{13.4–45}$$

The probability of error is simply 1 minus the probability that $U_1 > U_m$ for $m = 2, 3, \ldots, M$. Since the signals are orthogonal and the additive noise processes are mutually statistically independent, the random variables $U_1, U_2, \ldots, U_M$ are also mutually statistically independent. The probability density function of U_1 was given in Equation 13.4–31. On the other hand, $U_2, \ldots, U_M$ are identically distributed and described by the marginal probability density function in Equation 13.4–32. With U_1 fixed, the joint probability $P(U_2 < U_1, U_3 < U_1, \ldots, U_m < U_1)$ is equal to $P(U_2 < U_1)$ raised to the $M-1$ power. Now,

$$\begin{aligned} P(U_2 < U_1 \mid U_1 = u_1) &= \int_0^{u_1} p(u_2)\, du_2 \\ &= 1 - \exp\left(-\frac{u_1}{2\sigma_2^2}\right) \sum_{k=0}^{L-1} \frac{1}{k!} \left(\frac{u_1}{2\sigma_2^2}\right)^k \end{aligned} \tag{13.4–46}$$

where $\sigma_2^2 = 2\mathcal{E}N_0$. The $M-1$ power of this probability is then averaged over the probability density function of U_1 to yield the probability of a correct decision. If we subtract this result from unity, we obtain the probability of error in the form given by Hahn (1962)

$$\begin{aligned} P_e &= 1 - \int_0^\infty \frac{1}{\left(2\sigma_1^2\right)^L (L-1)!} u_1^{L-1} \exp\left(-\frac{u_1}{2\sigma_1^2}\right) \\ &\quad \times \left[1 - \exp\left(-\frac{u_1}{2\sigma_2^2}\right) \sum_{k=0}^{L-1} \frac{1}{k!} \left(\frac{u_1}{2\sigma_2^2}\right)^k\right]^{M-1} du_1 \\ &= 1 - \int_0^\infty \frac{1}{(1+\overline{\gamma}_c)^L (L-1)!} u_1^{L-1} \exp\left(-\frac{u_1}{1+\overline{\gamma}_c}\right) \\ &\quad \times \left(1 - e^{-u_1} \sum_{k=0}^{L-1} \frac{u_1^k}{k!}\right)^{M-1} du_1 \end{aligned} \tag{13.4–47}$$

where $\overline{\gamma}_c$ is the average SNR per diversity channel. The average SNR per bit is $\overline{\gamma}_b = L\overline{\gamma}_c / \log_2 M = L\overline{\gamma}_c / k$.

The integral in Equation 13.4–47 can be expressed in closed form as a double summation. This can be seen if we write

$$\left(\sum_{k=0}^{L-1} \frac{u_1^k}{k!}\right)^m = \sum_{k=0}^{m(L-1)} \beta_{km} u_1^k \tag{13.4–48}$$

where β_{km} is the set of coefficients in the above expansion. Then it follows that Equation 13.4–47 reduces to

$$\begin{aligned} P_e &= \frac{1}{(L-1)!} \sum_{m=1}^{M-1} \frac{(-1)^{m+1} \binom{M-1}{m}}{(1+m+m\overline{\gamma}_c)^L} \\ &\quad \times \sum_{k=0}^{m(L-1)} \beta_{km} (L-1+k)! \left(\frac{1+\overline{\gamma}_c}{1+m+m\overline{\gamma}_c}\right)^k \end{aligned} \tag{13.4–49}$$

When there is no diversity ($L = 1$), the error probability in Equation 13.4–49 reduces to the simple form

$$P_e = \sum_{m=1}^{M-1} \frac{(-1)^{m+1}\binom{M-1}{m}}{1 + m + m\overline{\gamma}_c} \tag{13.4–50}$$

The symbol error rate P_e may be converted to an equivalent bit error rate by multiplying P_e with $2^{k-1}/(2^k - 1)$.

Although the expression for P_e given in Equation 13.4–49 is in closed form, it is computationally cumbersome to evaluate for large values of M and L. An alternative is to evaluate P_M by numerical integration using the expression in Equation 13.4–47. The results illustrated in the following graphs were generated from Equation 13.4–47.

First of all, let us observe the error rate performance of M-ary orthogonal signaling with square-law combining as a function of the order of diversity. Figures 13.4–5 and 13.4–6 illustrate the characteristics of P_e for $M = 2$ and 4 as a function of L when the total SNR, defined as $\overline{\gamma}_t = L\overline{\gamma}_c$, remains fixed. These results indicate that there is an optimum order of diversity for each $\overline{\gamma}_t$. That is, for any $\overline{\gamma}_t$, there is a value of L for which P_e is a minimum. A careful observation of these graphs reveals that the minimum

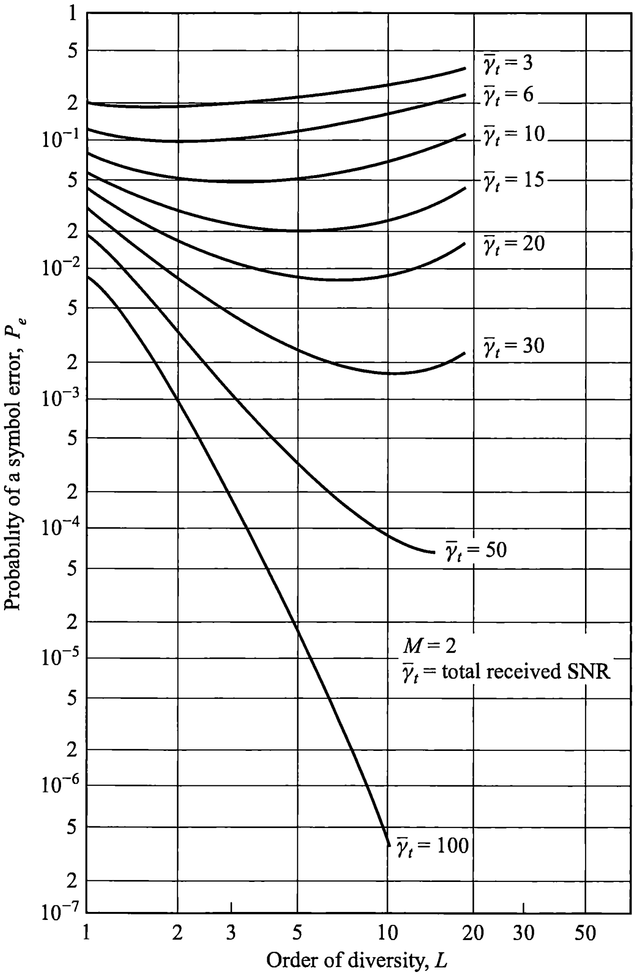


FIGURE 13.4–5
Performance of square-law-detected binary orthogonal signals as a function of diversity.

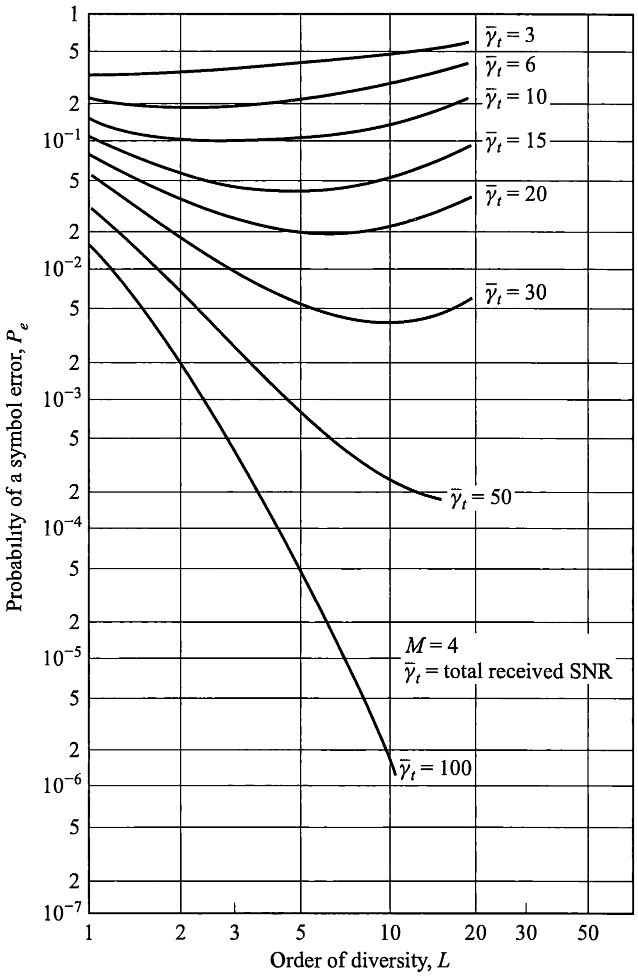


FIGURE 13.4–6
Performance of square-law-detected $M = 4$ orthogonal signals as a function of diversity.

in P_e is obtained when $\bar{\gamma}_c = \bar{\gamma}_t/L \approx 3$. This result appears to be independent of the alphabet size M.

Second, let us observe the error rate P_e as a function of the average SNR per bit, defined as $\bar{\gamma}_b = L\bar{\gamma}_c/k$. (If we interpret M-ary orthogonal FSK as a form of coding and the order of diversity as the number of times a symbol is repeated in a repetition code, then $\bar{\gamma}_b = \bar{\gamma}_c/R_c$, where $R_c = k/L$ is the code rate.) The graphs of P_e versus $\bar{\gamma}_b$ for $M = 2, 4, 8, 16, 32$ and $L = 1, 2, 4$ are shown in Figure 13.4–7. These results illustrate the gain in performance as M increases and L increases. First, we note that a significant gain in performance is obtained by increasing L. Second, we note that the gain in performance obtained with an increase in M is relatively small when L is small. However, as L increases, the gain achieved by increasing M also increases. Since an increase in either parameter results in an expansion of bandwidth, i.e.,

$$B_e = \frac{LM}{\log_2 M} \tag{13.4–51}$$

the results illustrated in Figure 13.4–7 indicate that an increase in L is more efficient than a corresponding increase in M. As we shall see in Chapter 14, coding is a bandwidth-effective means for obtaining diversity in the signal transmitted over the fading channel.

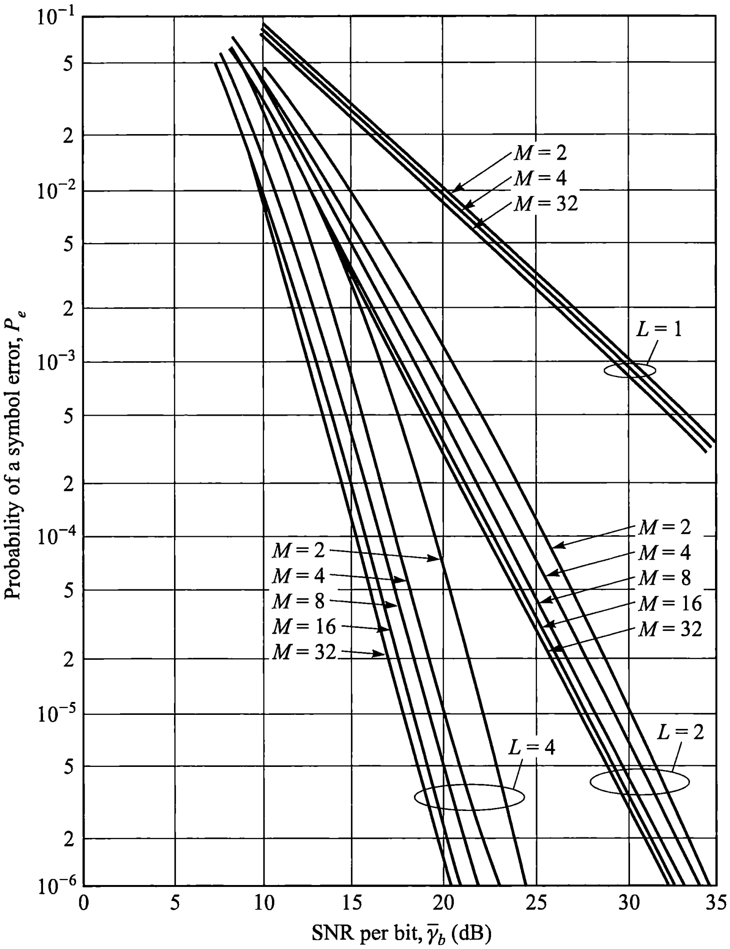


FIGURE 13.4–7
Performance of orthogonal signaling with M and L as parameters.

Chernov bound Before concluding this section, we develop a Chernov upper bound on the error probability of binary orthogonal signaling with Lth-order diversity, which will be useful in our discussion of coding for fading channels, the topic of Chapter 14. Our starting point is the expression for the two decision variables U_1 and U_2 given by Equation 13.4–29, where U_1 consists of the square-law-combined signal-plus-noise terms and U_2 consists of square-law-combined noise terms. The binary probability of error, denoted here by $P_b(L)$, is

$$\begin{aligned} P_b(L) &= P(U_2 - U_1 > 0) \\ &= P(X > 0) = \int_0^\infty p(x)\,dx \end{aligned} \qquad (13.4\text{–}52)$$

where the random variable X is defined as

$$X = U_2 - U_1 = \sum_{k=1}^{L} \left(|N_{k2}|^2 - |2\mathcal{E}\alpha_k + N_{k1}|^2 \right) \qquad (13.4\text{–}53)$$

The phase terms $\{\phi_k\}$ in U_1 have been dropped since they do not affect the performance of the square-law detector.

Using the Chernov bound, the error probability in 13.4–52 can be expressed in the form

$$P_b(L) \leq E(e^{\zeta X}) \tag{13.4–54}$$

where the parameter $\zeta > 0$ is optimized to yield a tight bound. Upon substituting for the random variable X from Equation 13.4–53 and noting that the random variables in the summation are mutually statistically independent, we obtain the result

$$P_b(L) \leq \prod_{k=1}^{L} E\left(e^{\zeta|N_{k2}|^2}\right) E\left(e^{-\zeta|2\mathcal{E}\alpha_k+N_{k1}|^2}\right) \tag{13.4–55}$$

But

$$E\left(e^{\zeta|N_{k2}|^2}\right) = \frac{1}{1-2\zeta\sigma_2^2}, \qquad \zeta < \frac{1}{2\sigma_2^2} \tag{13.4–56}$$

and

$$E\left(e^{-\zeta|2\mathcal{E}\alpha_k+N_{k1}|^2}\right) = \frac{1}{1+2\zeta\sigma_1^2}, \qquad \zeta > \frac{-1}{2\sigma_1^2} \tag{13.4–57}$$

where $\sigma_2^2 = 2\mathcal{E}N_0$, $\sigma_1^2 = 2\mathcal{E}N_0(1+\overline{\gamma}_c)$, and $\overline{\gamma}_c$ is the average SNR per diversity channel. Note that σ_1^2 and σ_2^2 are independent of k, i.e., the additive noise terms on the L diversity channels as well as the fading statistics are identically distributed. Consequently, Equation 13.4–55 reduces to

$$P_b(L) \leq \left[\frac{1}{\left(1-2\zeta\sigma_2^2\right)\left(1+2\zeta\sigma_1^2\right)}\right]^L, \qquad 0 \leq \zeta \leq \frac{1}{2\sigma_2^2} \tag{13.4–58}$$

By differentiating the right-hand side of Equation 13.4–58 with respect to ζ, we find that the upper bound is minimized when

$$\zeta = \frac{\sigma_1^2 - \sigma_2^2}{4\sigma_1^2\sigma_2^2} \tag{13.4–59}$$

Substitution of Equation 13.4–59 for ζ into Equation 13.4–58 yields the Chernov upper bound in the form

$$P_b(L) \leq \left[\frac{4(1+\overline{\gamma}_c)}{(2+\overline{\gamma}_c)^2}\right]^L \tag{13.4–60}$$

It is interesting to note that Equation 13.4–60 may also be expressed as

$$P_b(L) \leq [4p(1-p)]^L \tag{13.4–61}$$

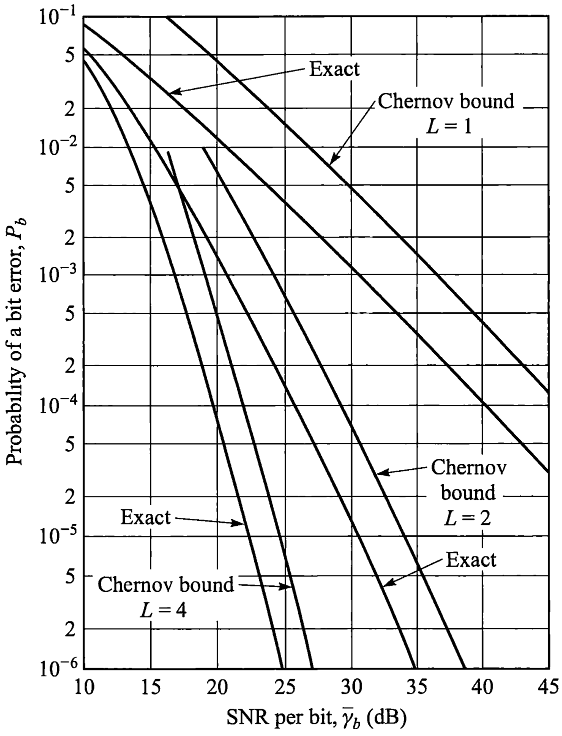


FIGURE 13.4–8
Comparison of Chernov bound with exact error probability.

where $p = 1/(2 + \bar{\gamma}_c)$ is the probability of error for binary orthogonal signaling on a fading channel without diversity.

A comparison of the Chernov bound in Equation 13.4–60 with the exact error probability for binary orthogonal signaling and square-law combining of the L diversity signals, which is given by the expression

$$\begin{aligned} P_b(L) &= \left(\frac{1}{2+\bar{\gamma}_c}\right)^L \sum_{k=0}^{L-1} \binom{L-1+k}{k} \left(\frac{1+\bar{\gamma}_c}{2+\bar{\gamma}_c}\right)^k \\ &= p^L \sum_{k=0}^{L-1} \binom{L-1+k}{k} (1-p)^k \end{aligned} \tag{13.4–62}$$

reveals the tightness of the bound. Figure 13.4–8 illustrates this comparison. We observe that the Chernov upper bound is approximately 6 dB from the exact error probability for $L = 1$, but, as L increases, it becomes tighter. For example, the difference between the bound and the exact error probability is about 2.5 dB when $L = 4$.

Finally we mention that the error probability for M-ary orthogonal signaling with diversity can be upper-bounded by means of the union bound

$$P_e \leq (M-1)P_2(L) \tag{13.4–63}$$

where we may use either the exact expression given in Equation 13.4–62 or the Chernov bound in Equation 13.4–60 for $P_b(L)$.

13.5
SIGNALING OVER A FREQUENCY-SELECTIVE, SLOWLY FADING CHANNEL: THE RAKE DEMODULATOR

When the spread factor of the channel satisfies the condition $T_m B_d \ll 1$, it is possible to select signals having a bandwidth $W \ll (\Delta f)_c$ and a signal duration $T \ll (\Delta t)_c$. Thus, the channel is frequency-nonselective and slowly fading. In such a channel, diversity techniques can be employed to overcome the severe consequences of fading.

When a bandwidth $W \gg (\Delta f)_c$ is available to the user, the channel can be subdivided into a number of frequency-division multiplexed (FDM) subchannels having a mutual separation in center frequencies of at least $(\Delta f)_c$. Then the same signal can be transmitted on the FDM subchannels, and, thus, frequency diversity is obtained. In this section, we describe an alternative method.

13.5–1 A Tapped-Delay-Line Channel Model

As we shall now demonstrate, a more direct method for achieving basically the same results is to employ a wideband signal covering the bandwidth W. The channel is still assumed to be slowly fading by virtue of the assumption that $T \ll (\Delta t)_c$. Now suppose that W is the bandwidth occupied by the real band-pass signal. Then the band occupancy of the equivalent low-pass signal $s_l(t)$ is $|f| \le \frac{1}{2}W$. Since $s_l(t)$ is band-limited to $|f| \le \frac{1}{2}W$, application of the sampling theorem results in the signal representation

$$s_l(t) = \sum_{n=-\infty}^{\infty} s_l\left(\frac{n}{W}\right) \frac{\sin[\pi W(t - n/W)]}{\pi W(t - n/W)} \tag{13.5–1}$$

The Fourier transform of $s_l(t)$ is

$$S_l(f) = \begin{cases} \dfrac{1}{W} \displaystyle\sum_{n=-\infty}^{\infty} s_l(n/W) e^{-j2\pi fn/W} & |f| \le \frac{1}{2}W \\ 0 & |f| > \frac{1}{2}W \end{cases} \tag{13.5–2}$$

The noiseless received signal from a frequency-selective channel was previously expressed in the form

$$r_l(t) = \int_{-\infty}^{\infty} C(f; t) S_l(f) e^{j2\pi ft}\, df \tag{13.5–3}$$

where $C(f;t)$ is the time-variant transfer function. Substitution for $S_l(f)$ from Equation 13.5–2 into 13.5–3 yields

$$\begin{aligned} r_l(t) &= \frac{1}{W}\sum_{n=-\infty}^{\infty} s_l(n/W)\int_{-\infty}^{\infty} C(f;t)e^{j2\pi f(t-n/W)}\,df \\ &= \frac{1}{W}\sum_{n=-\infty}^{\infty} s_l(n/W)c(t-n/W;t) \end{aligned} \tag{13.5–4}$$

where $c(\tau;t)$ is the time-variant impulse response. We observe that Equation 13.5–4 has the form of a convolution sum. Hence, it can also be expressed in the alternative form

$$r_l(t) = \frac{1}{W}\sum_{n=-\infty}^{\infty} s_l(t-n/W)c(n/W;t) \tag{13.5–5}$$

It is convenient to define a set of time-variable channel coefficients as

$$c_n(t) = \frac{1}{W}c\left(\frac{n}{W};t\right) \tag{13.5–6}$$

Then Equation 13.5–5 expressed in terms of these channel coefficients becomes

$$r_l(t) = \sum_{n=-\infty}^{\infty} c_n(t)s_l(t-n/W) \tag{13.5–7}$$

The form for the received signal in Equation 13.5–7 implies that the time-variant frequency-selective channel can be modeled or represented as a tapped delay line with tap spacing $1/W$ and tap weight coefficients $\{c_n(t)\}$. In fact, we deduce from Equation 13.5–7 that the low-pass impulse response for the channel is

$$c(\tau;t) = \sum_{n=-\infty}^{\infty} c_n(t)\delta(\tau-n/W) \tag{13.5–8}$$

and the corresponding time-variant transfer function is

$$C(f;t) = \sum_{n=-\infty}^{\infty} c_n(t)e^{-j2\pi fn/W} \tag{13.5–9}$$

Thus, with an equivalent low-pass-signal having a bandwidth $\frac{1}{2}W$, where $W \gg (\Delta f)_c$, we achieve a resolution of $1/W$ in the multipath delay profile. Since the total multipath spread is T_m, for all practical purposes the tapped delay line model for the channel can be truncated at $L = \lfloor T_m W \rfloor + 1$ taps. Then the noiseless received signal can be expressed in the form

$$r_l(t) = \sum_{n=1}^{L} c_n(t)s_l\left(t-\frac{n}{W}\right) \tag{13.5–10}$$

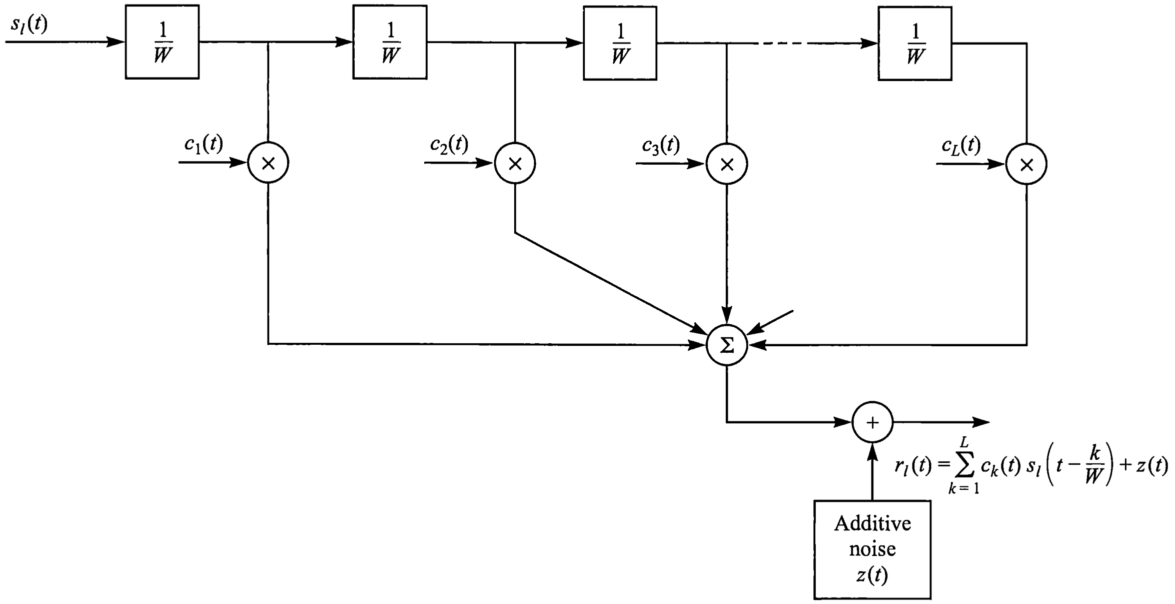


FIGURE 13.5–1
Trapped delay line model of frequency-selective channel.

The truncated tapped delay line model is shown in Figure 13.5–1. In accordance with the statistical characterization of the channel presented in Section 13.1, the time-variant tap weights $\{c_n(t)\}$ are complex-valued stationary random processes. In the special case of Rayleigh fading, the magnitudes $|c_n(t)| \equiv \alpha_n(t)$ are Rayleigh-distributed and the phases $\phi_n(t)$ are uniformly distributed. Since the $\{c_n(t)\}$ represent the tap weights corresponding to the L different delays $\tau = n/W, n = 1, 2, \ldots, L$, the uncorrelated scattering assumption made in Section 13.1 implies that the $\{c_n(t)\}$ are mutually uncorrelated. When the $\{c_n(t)\}$ are Gaussian random processes, they are statistically independent.

13.5–2 The RAKE Demodulator

We now consider the problem of digital signaling over a frequency-selective channel that is modeled by a tapped delay line with statistically independent time-variant tap weights $\{c_n(t)\}$. It is apparent at the outset, however, that the tapped delay line model with statistically independent tap weights provides us with L replicas of the same transmitted signal at the receiver. Hence, a receiver that processes the received signal in an optimum manner will achieve the performance of an equivalent Lth-order diversity communication system.

Let us consider binary signaling over the channel. We have two equal-energy signals $s_{l1}(t)$ and $s_{l2}(t)$, which are either antipodal or orthogonal. Their time duration T is selected to satisfy the condition $T \gg T_m$. Thus, we may neglect any intersymbol interference due to multipath. Since the bandwidth of the signal exceeds the coherent bandwidth of the channel, the received signal is expressed as

$$\begin{aligned} r_l(t) &= \sum_{k=1}^{L} c_k(t) s_{li}(t - k/W) + z(t) \\ &= v_i(t) + z(t), \qquad 0 \le t \le T, \qquad i = 1, 2 \end{aligned} \tag{13.5–11}$$

where $z(t)$ is a complex-valued zero-mean white Gaussian noise process. Assume for the moment that the channel tap weights are known. Then the optimum demodulator consists of two filters matched to $v_1(t)$ and $v_2(t)$. The demodulator output is sampled at the symbol rate and the samples are passed to a decision circuit that selects the signal corresponding to the largest output. An equivalent optimum demodulator employs cross correlation instead of matched filtering. In either case, the decision variables for coherent detection of the binary signals can be expressed as

$$\begin{aligned} U_m &= \operatorname{Re}\left[\int_0^T r_l(t) v_m^*(t)\, dt\right] \\ &= \operatorname{Re}\left[\sum_{k=1}^{L} \int_0^T r_l(t) c_k^*(t) s_m^*(t - k/W)\, dt\right], \qquad m = 1, 2 \end{aligned} \tag{13.5–12}$$

Figure 13.5–2 illustrates the operations involved in the computation of the decision variables. In this realization of the optimum receiver, the two reference signals are delayed and correlated with the received signal $r_l(t)$.

An alternative realization of the optimum demodulator employs a single delay line through which is passed the received signal $r_l(t)$. The signal at each tap is correlated with $c_k^*(t) s_{lm}^*(t)$, where $k = 1, 2, \ldots, L$ and $m = 1, 2$. This receiver structure is shown in Figure 13.5–3. In effect, the tapped delay line demodulator attempts to collect the signal energy from all the received signal paths that fall within the span of the delay line and carry the same information. Its action is somewhat analogous to an ordinary garden rake and, consequently, the name "RAKE demodulator" has been coined for this demodulator structure by Price and Green (1958). The taps on the RAKE demodulator are often called "RAKE fingers."

13.5–3 Performance of RAKE Demodulator

We shall now evaluate the performance of the RAKE demodulator under the condition that the fading is sufficiently slow to allow us to estimate $c_k(t)$ perfectly (without noise). Furthermore, within any one signaling interval, $c_k(t)$ is treated as a constant and denoted as c_k. Thus the decision variables in Equation 13.5–12 may be expressed in the form

$$U_m = \operatorname{Re}\left[\sum_{k=1}^{L} c_k^* \int_0^T r(t) s_{lm}^*(t - k/W)\, dt\right], \qquad m = 1, 2 \tag{13.5–13}$$

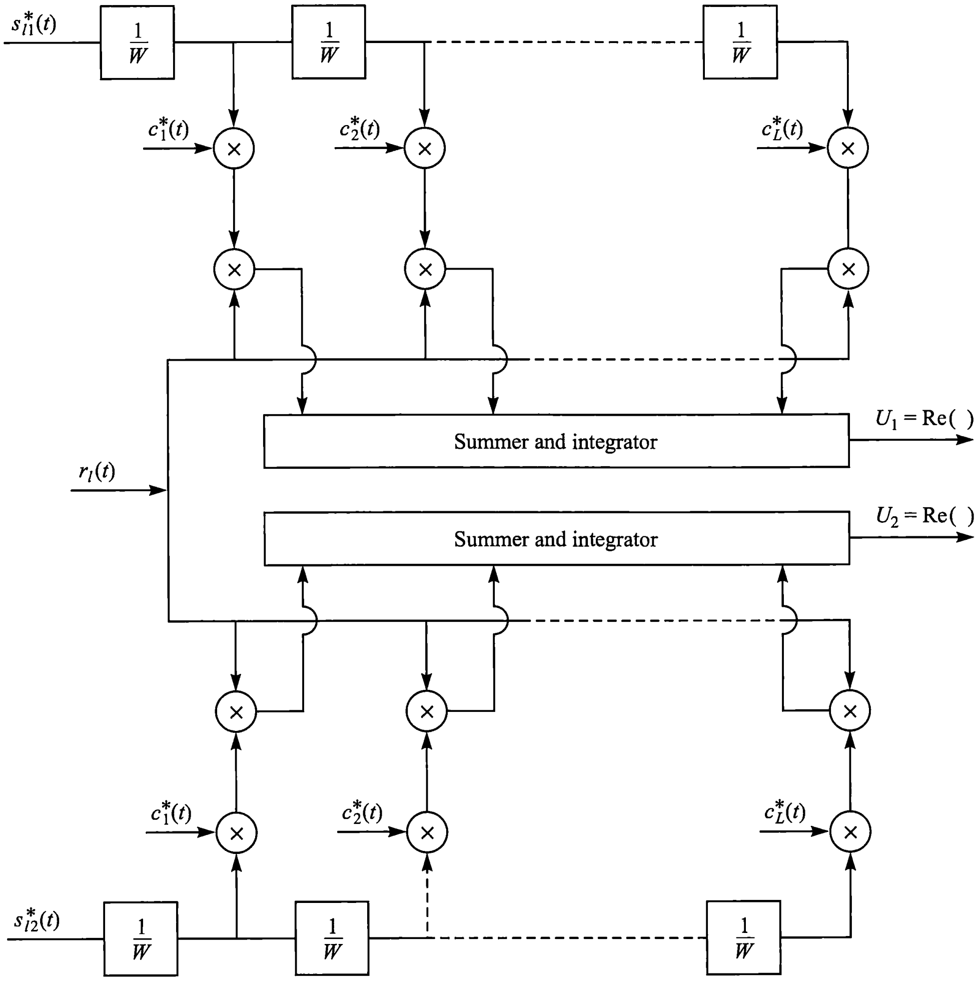

$s_{l1}^*(t)$
$\frac{1}{W}$
$c_1^*(t)$
$c_2^*(t)$
$c_L^*(t)$
$r_l(t)$
Summer and integrator
$U_1 = \text{Re}(\ )$
Summer and integrator
$U_2 = \text{Re}(\ )$
$s_{l2}^*(t)$

FIGURE 13.5–2
Optimum demodulator for wideband binary signals (delayed reference configuration).

Suppose the transmitted signal is $s_{l1}(t)$; then the received signal is

$$r_l(t) = \sum_{n=1}^{L} c_n s_{l1}(t - n/W) + z(t), \qquad 0 \le t \le T \tag{13.5–14}$$

Substitution of Equation 13.5–14 into Equation 13.5–13 yields

$$\begin{aligned} U_m &= \text{Re}\left[\sum_{k=1}^{L} c_k^* \sum_{n=1}^{L} c_n \int_0^T s_{l1}(t - n/W) s_{lm}^*(t - k/W)\, dt\right] \\ &\quad + \text{Re}\left[\sum_{k=1}^{L} c_k^* \int_0^T z(t) s_{lm}^*(t - k/W)\, dt\right], \qquad m = 1, 2 \end{aligned} \tag{13.5–15}$$

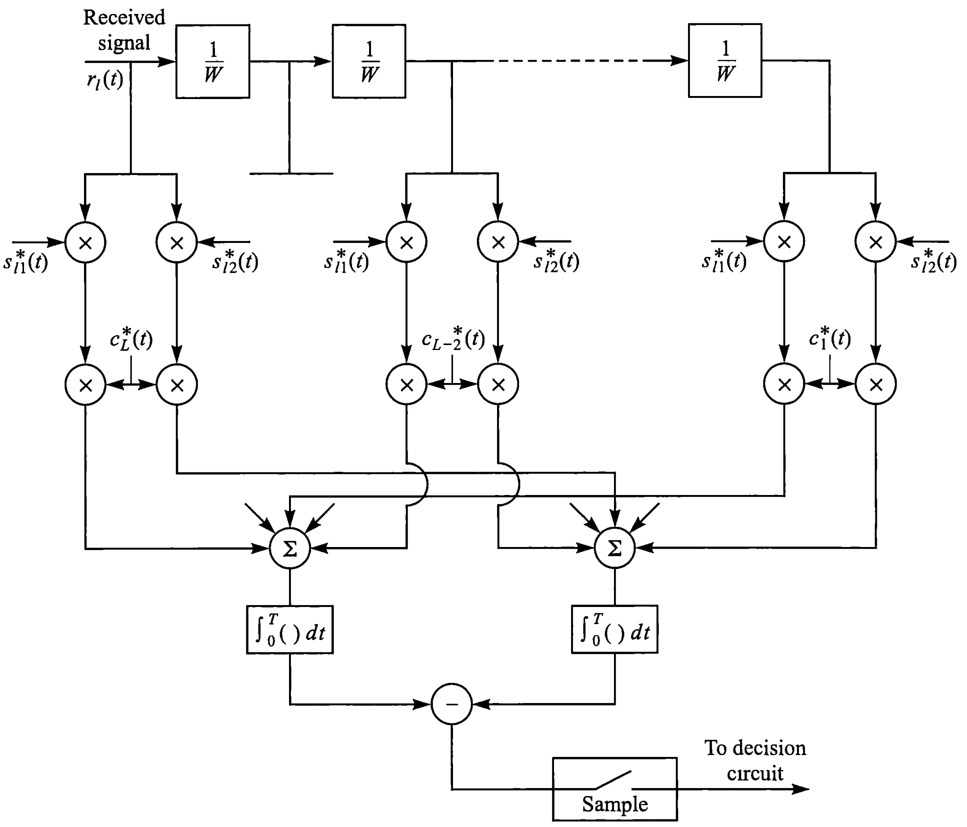


FIGURE 13.5–3

Optimum demodulator for wideband binary signals (delayed received signal configuration).

Usually the wideband signals $s_{l1}(t)$ and $s_{l2}(t)$ are generated from pseudorandom sequences, which result in signals that have the property

$$\int_0^T s_{li}(t-n/W)s_{li}^*(t-k/W)\,dt \approx 0, \qquad k \neq n, \qquad i = 1, 2 \tag{13.5–16}$$

If we assume that our binary signals are designed to satisfy this property, then Equation 13.5–15 simplifies to[†]

$$\begin{aligned} U_m &= \operatorname{Re}\left[\sum_{k=1}^{L} |c_k|^2 \int_0^T s_{l1}(t-k/W)s_{lm}^*(t-k/W)\,dt\right] \\ &\quad + \operatorname{Re}\left[\sum_{k=1}^{L} c_k^* \int_0^T z(t)s_{lm}^*(t-k/W)\,dt\right], \qquad m = 1, 2 \end{aligned} \tag{13.5–17}$$

[†] Although the orthogonality property specified by Equation 13.5–16 can be satisfied by proper selection of the pseudorandom sequences, the cross correlation of $s_{l1}(t-n/W)$ with $s_{li}^*(t-k/W)$ gives rise to a signal-dependent self-noise, which ultimately limits the performance. For simplicity, we do not consider the self-noise term in the following calculations. Consequently, the performance results presented below should be considered as lower bounds (ideal RAKE). An approximation to the performance of the RAKE can be obtained by treating the self-noise as an additional Gaussian noise component with noise power equal to its variance.

When the binary signals are antipodal, a single decision variable suffices. In this case, Equation 13.5–17 reduces to

$$U_1 = \text{Re}\left(2\mathcal{E}\sum_{k=1}^{L}\alpha_k^2 + \sum_{k=1}^{L}\alpha_k N_k\right) \tag{13.5–18}$$

where $\alpha_k = |c_k|$ and

$$N_k = e^{-j\phi_k}\int_0^T z(t)s_l^*(t - k/W)\,dt \tag{13.5–19}$$

But Equation 13.5–18 is identical to the decision variable given in Equation 13.4–4, which corresponds to the output of a maximal ratio combiner in a system with Lth-order diversity. Consequently, the RAKE demodulator with perfect (noiseless) estimates of the channel tap weights is equivalent to a maximal ratio combiner in a system with Lth-order diversity. Thus, when all the tap weights have the same mean-square value, i.e., $E(\alpha_k^2)$ is the same for all k, the error rate performance of the RAKE demodulator is given by Equations 13.4–15 and 13.4–16. On the other hand, when the mean-square values $E(\alpha_k^2)$ are not identical for all k, the derivation of the error rate performance must be repeated since Equation 13.4–15 no longer applies.

We shall derive the probability of error for binary antipodal and orthogonal signals under the condition that the mean-square values of $\{\alpha_k\}$ are distinct. We begin with the conditional error probability

$$P_b(\gamma_b) = Q\left(\sqrt{\gamma_b(1-\rho_r)}\right) \tag{13.5–20}$$

where $\rho_r = -1$ for antipodal signals, $\rho_r = 0$ for orthogonal signals, and

$$\gamma_b = \frac{\mathcal{E}}{N_0}\sum_{k=1}^{L}\alpha_k^2 = \sum_{k=1}^{L}\gamma_k \tag{13.5–21}$$

Each of the $\{\gamma_k\}$ is distributed according to a chi-squared distribution with two degrees of freedom. That is,

$$p(\gamma_k) = \frac{1}{\overline{\gamma}_k}e^{-\gamma_k/\overline{\gamma}_k} \tag{13.5–22}$$

where $\overline{\gamma}_k$ is the average SNR for the kth path, defined as

$$\overline{\gamma}_k = \frac{\mathcal{E}}{N_0}E\left(\alpha_k^2\right) \tag{13.5–23}$$

Furthermore, from Equation 13.4–10 we know that the characteristic function of γ_k is

$$\Phi_{\gamma_k}(v) = \frac{1}{1 - jv\overline{\gamma}_k} \tag{13.5–24}$$

Since γ_b is the sum of L statistically independent components $\{\gamma_k\}$, the characteristic function of γ_b is

$$\Phi_{\gamma_b}(v) = \prod_{k=1}^{L} \frac{1}{1 - jv\overline{\gamma}_k} \tag{13.5–25}$$

The inverse Fourier transform of the characteristic function in Equation 13.5–25 yields the probability density function of γ_b in the form

$$p(\gamma_b) = \sum_{k=1}^{L} \frac{\pi_k}{\overline{\gamma}_k} e^{-\gamma_b/\overline{\gamma}_k}, \qquad \gamma_b \geq 0 \tag{13.5–26}$$

where π_k is defined as

$$\pi_k = \prod_{\substack{i=1 \\ i\neq k}}^{L} \frac{\overline{\gamma}_k}{\overline{\gamma}_k - \overline{\gamma}_i} \tag{13.5–27}$$

When the conditional error probability in Equation 13.5–20 is averaged over the probability density function given in Equation 13.5–26, the result is

$$P_b = \tfrac{1}{2} \sum_{k=1}^{L} \pi_k \left[1 - \sqrt{\frac{\overline{\gamma}_k(1-\rho_r)}{2 + \overline{\gamma}_k(1-\rho_r)}} \right] \tag{13.5–28}$$

This error probability can be approximated as ($\overline{\gamma}_k \gg 1$)

$$P_b \approx \binom{2L-1}{L} \prod_{k=1}^{L} \frac{1}{2\overline{\gamma}_k(1-\rho_r)} \tag{13.5–29}$$

By comparing Equation 13.5–29 for $\rho_r = -1$ with Equation 13.4–18, we observe that the same type of asymptotic behavior is obtained for the case of unequal SNR per path and the case of equal SNR per path.

In the derivation of the error rate performance of the RAKE demodulator, we assumed that the estimates of the channel tap weights are perfect. In practice, relatively good estimates can be obtained if the channel fading is sufficiently slow, e.g., $(\Delta t)_c/T \geq 100$, where T is the signaling interval. Figure 13.5–4 illustrates a method for estimating the tap weights when the binary signaling waveforms are orthogonal. The estimate is the output of the low-pass filter at each tap. At any one instant in time, the incoming signal is either $s_{l1}(t)$ or $s_{l2}(t)$. Hence, the input to the low-pass filter used to estimate $c_k(t)$ contains signal plus noise from one of the correlators and noise only from the other correlator. This method for channel estimation is not appropriate for antipodal signals, because the addition of the two correlator outputs results in signal cancellation. Instead, a single correlator can be employed for antipodal signals. Its output is fed to the input of the low-pass filter after the information-bearing signal is removed. To accomplish this, we must introduce a delay of one signaling interval into the channel estimation procedure, as illustrated in Figure 13.5–5. That is, first the receiver must decide whether the information in the received signal is $+1$ or -1 and, then, it uses the

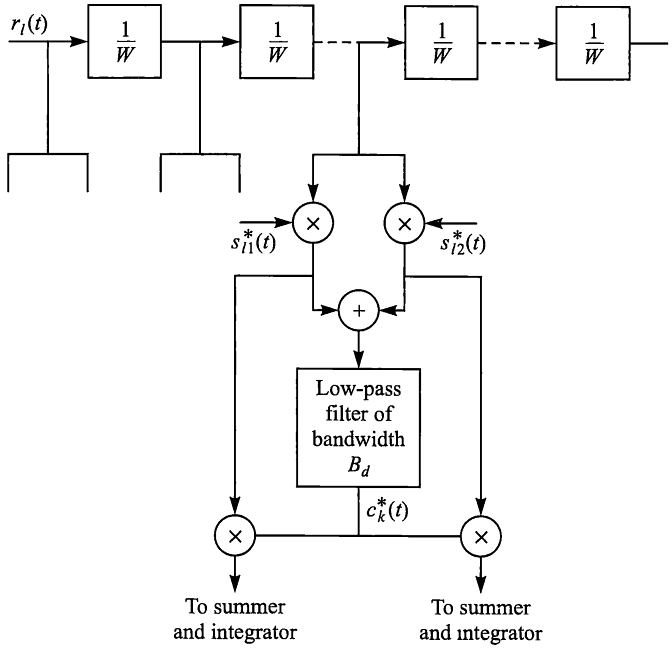


FIGURE 13.5–4
Channel tap weight estimation with binary orthogonal signals.

decision to remove the information from the correlator ouput prior to feeding it to the low-pass filter.

If we choose not to estimate the tap weights of the frequency-selective channel, we may use either DPSK signaling or noncoherently detected orthogonal signaling. The RAKE demodulator structure for DPSK is illustrated in Figure 13.5–6. It is apparent that when the transmitted signal waveform $s_l(t)$ satisfies the orthogonality property given in Equation 13.5–16, the decision variable is identical to that given in Equation 13.4–23 for an Lth-order diversity system. Consequently, the error rate performance of the RAKE demodulator for a binary DPSK is identical to that given in Equation 13.4–15 with $\mu = \overline{\gamma}_c/(1+\overline{\gamma}_c)$, when all the signal paths have the same SNR $\overline{\gamma}_c$. On the other hand, when the SNRs $\{\overline{\gamma}_k\}$ are distinct, the error probability can be obtained by averaging Equation 13.4–24, which is the probability of error conditioned on a time-invariant channel, over the probability density function of γ_b given by Equation 13.5–26. The result of this integration is

$$P_b = \left(\tfrac{1}{2}\right)^{2L-1} \sum_{m=0}^{L-1} m! b_m \sum_{k=1}^{L} \frac{\pi_k}{\overline{\gamma}_k} \left(\frac{\overline{\gamma}_k}{1+\overline{\gamma}_k}\right)^{m+1} \tag{13.5–30}$$

where π_k is defined in Equation 13.5–27 and b_m in Equation 13.4–25.

Finally, we consider binary orthognal signaling over the frequency-selective channel with square-law detection at the receiver. This type of signal is appropriate when the fading is rapid enough to preclude a good estimate of the channel tap weights. The RAKE demodulator with square-law combining of the signal from each tap is illustrated in Figure 13.5–7. In computing its performance, we again assume that the orthogonality property given in Equation 13.5–16 holds. Then the decision variables at

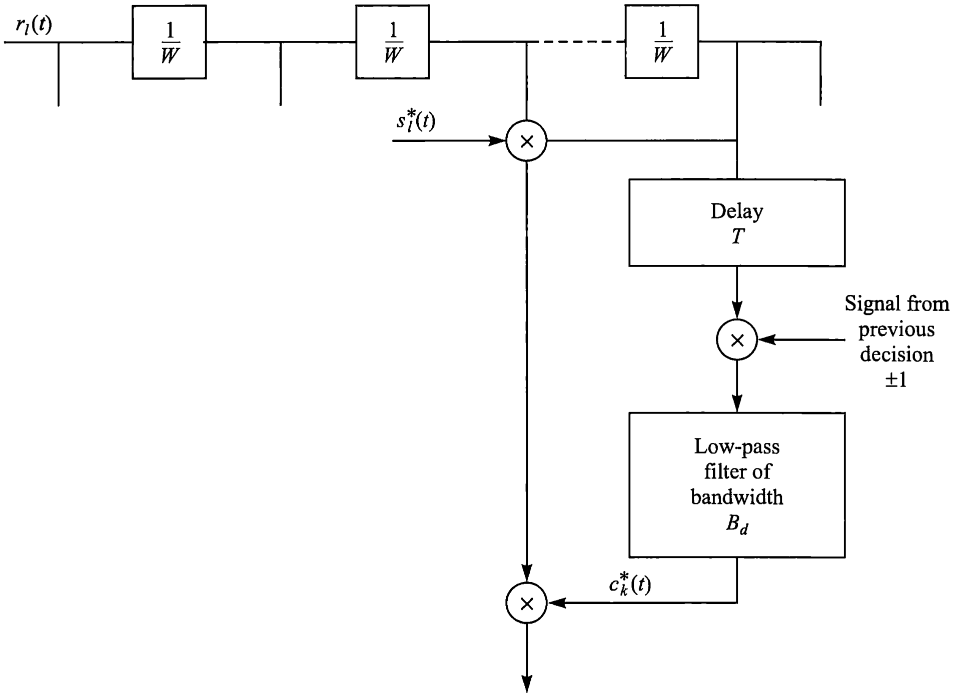

$r_l(t)$
$\frac{1}{W}$
$\frac{1}{W}$
$\frac{1}{W}$
$s_l^*(t)$
Delay T
Signal from previous decision ±1
Low-pass filter of bandwidth B_d
$c_k^*(t)$

FIGURE 13.5–5
Channel tap weight estimation with binary antipodal signals.

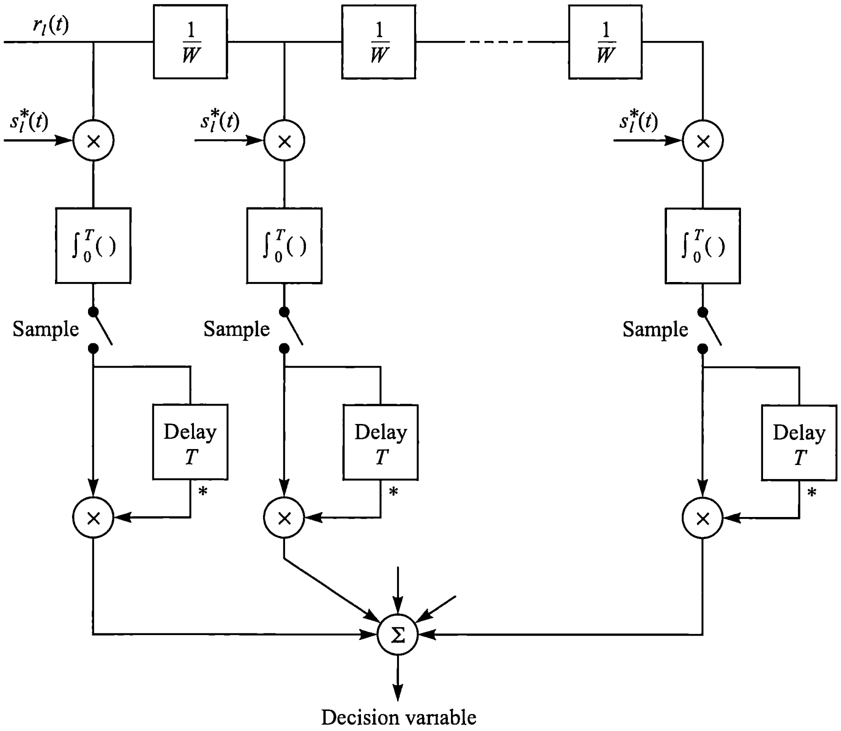

$r_l(t)$
$\frac{1}{W}$
$\frac{1}{W}$
$\frac{1}{W}$
$s_l^*(t)$
$s_l^*(t)$
$s_l^*(t)$
$\int_0^T()$
$\int_0^T()$
$\int_0^T()$
Sample
Sample
Sample
Delay T
Delay T
Delay T
*
*
*
Σ
Decision variable

FIGURE 13.5–6
RAKE demodulator for DPSK signals.

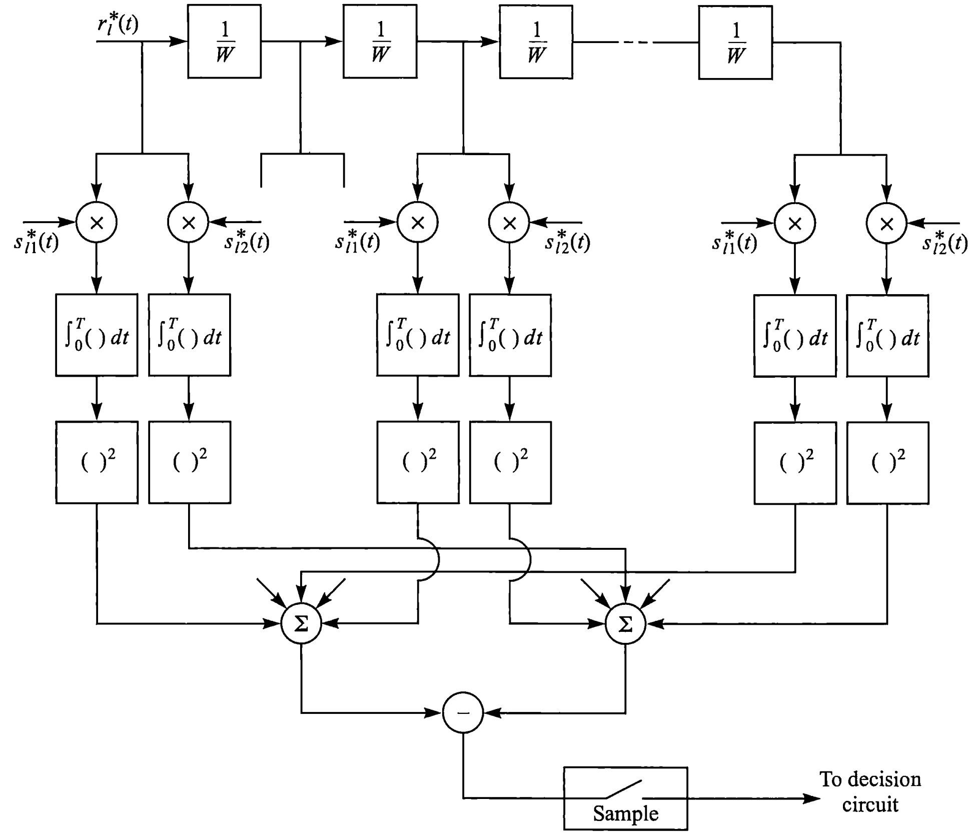


FIGURE 13.5–7
RAKE demodulator for square-law combination of orthogonal signals.

the output of the RAKE are

$$U_1 = \sum_{k=1}^{L} |2\mathcal{E}c_k + N_{k1}|^2 \qquad U_2 = \sum_{k=1}^{L} |N_{k2}|^2 \tag{13.5–31}$$

where we have assumed that $s_{l1}(t)$ was the transmitted signal. Again we observe that the decision variables are identical to the ones given in Equation 13.4–29, which apply to orthogonal signals with Lth-order diversity. Therefore, the performance of the RAKE demodulator for square-law-detected orthogonal signals is given by Equation 13.4–15 with $\mu = \bar{\gamma}_c/(2+\bar{\gamma}_c)$ when all the signal paths have the same SNR. If the SNRs are distinct, we can average the conditional error probability given by Equation 13.4–24, with γ_b replaced by $\frac{1}{2}\gamma_b$, over the probability density function $p(\gamma_b)$ given in Equation 13.5–26. The result of this averaging is given by Equation 13.5–30, with $\bar{\gamma}_k$ replaced by $\frac{1}{2}\bar{\gamma}_k$.

In the above analysis, the RAKE demodulator shown in Figure 13.5–7 for square-law combining of orthogonal signals is assumed to contain a signal component at each delay. If that is not the case, its performance will be degraded, since some of the tap correlators will contribute only noise. Under such conditions, the low-level, noise-only contributions from the tap correlators should be excluded from the combiner, as shown by Chyi et al. (1988).

The configurations of the RAKE demodulator presented in this section can be easily generalized to multilevel signaling. In fact, if M-ary PSK or DPSK is chosen, the RAKE structures presented in this section remain unchanged. Only the PSK and DPSK detectors that follow the RAKE correlator are different.

Generalized RAKE Demodulator

The RAKE demodulator described above is the optimum demodulator when the additive noise is white and Gaussian. However, there are communication scenarios in which additive interference from other users of the channel results in colored additive noise. This is the case, for example, in the downlink of a cellular communication system employing CDMA as a multiple access method. In this case, the spread spectrum signals transmitted from a base station to the mobile receivers carry information on synchronously transmitted orthogonal spreading codes. However, in transmission over a frequency-selective channel, the orthogonality of the code sequences is destroyed by the channel time dispersion due to multipath. As a consequence, the RAKE demodulator for any given mobile receiver must demodulate its desired signal in the presence of additional additive interference resulting from the cross-correlations of its desired spreading code sequence with the multipath corrupted code sequences that are assigned to the other mobile users. This additional interference is generally characterized as colored Gaussian noise, as shown by Bottomley (1993) and Klein (1997).

A model for the downlink transmission in a CDMA cellular communication system is illustrated in Figure 13.5–8. The base station transmits the combined signal.

$$s(t) = \sum_{k=1}^{K} s_k(t) \tag{13.5–32}$$

to the K mobile terminals, where each $s_k(t)$ is a spread spectrum signal intended for the kth user and the corresponding spreading code for the kth user is orthogonal with each of the spreading codes of the other $K - 1$ users. We assume that the signals propagate through a channel characterized by the baseband equivalent lowpass, time-invariant

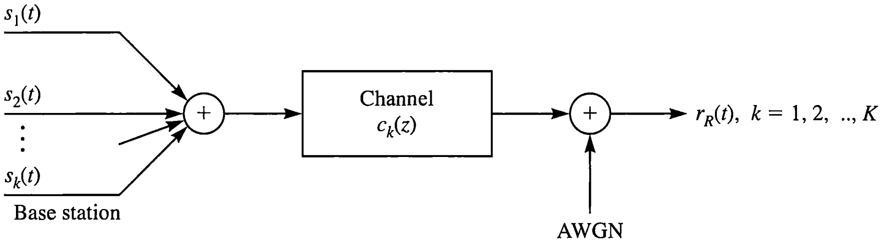


FIGURE 13.5–8
Model for the downlink transmission of a CDMA cellular communication system.

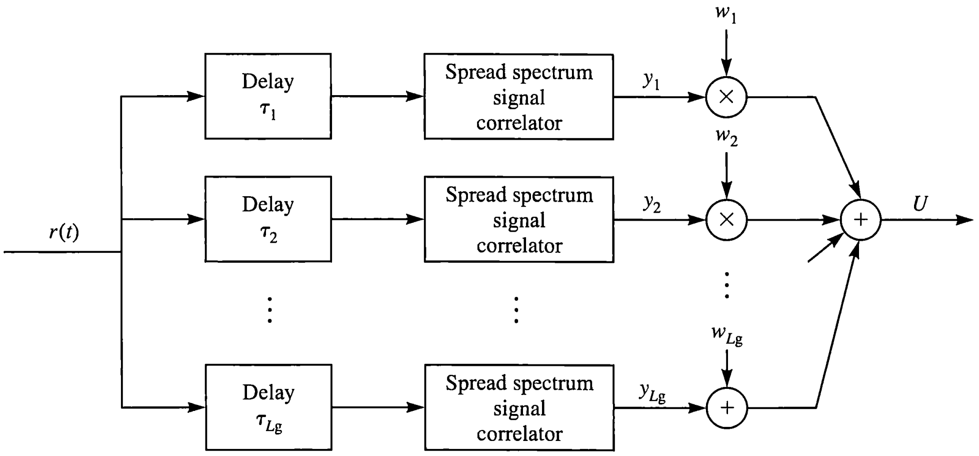

w_1
Delay
τ_1
Spread spectrum signal correlator
y_1
×
w_2
Delay
τ_2
Spread spectrum signal correlator
y_2
×
+
U
$r(t)$
w_{Lg}
Delay
τ_{Lg}
Spread spectrum signal correlator
y_{Lg}
+

FIGURE 13.5–9
Structure of generalized RAKE demodulator.

impulse response

$$c_k(\tau) = \sum_{i=1}^{L_k} c_{ki}\,\delta(\tau - \tau_{ki}), \qquad k = 1, 2, \ldots, K \tag{13.5–33}$$

where L_k is the number of resolvable multipath components, $\{c_{ki}\}$ are the complex-valued coefficients, and $\{\tau_{ki}\}$ are the corresponding time delays. To simplify this presentation, we focus on the processing at the receiver of the first user ($k = 1$) and drop the index k. In a CDMA cellular system, an unmodulated spread spectrum signal, say $s_0(t)$, is transmitted along with the information-bearing signals and serves as a pilot signal that is used by each mobile receiver to estimate the channel coefficients $\{c_i\}$ and the time delays $\{\tau_i\}$.

A conventional RAKE demodulator would consist of L "fingers" with each finger corresponding to one of the L channel delays, and the weights at the L fingers would be $\{c_i^*\}$, the complex conjugates of the corresponding channel coefficients. In contrast, a generalized RAKE demodulator consists of $L_g > L$ RAKE fingers, and the weights at the L_g fingers, denoted as $\{w_i\}$, are different from $\{c_i^*\}$. The structure of the generalized RAKE demodulator is illustrated in Figure 13.5–9 for phase coherent modulation such as PSK or QAM. The decision variable U at the detector may be expressed as

$$U = \boldsymbol{w}^H \boldsymbol{y} \tag{13.5–34}$$

It is convenient to express the received vector $\boldsymbol{y}$ at the output of the cross-correlators as

$$\boldsymbol{y} = \boldsymbol{g}b + \boldsymbol{z} \tag{13.5–35}$$

where $\boldsymbol{g}$ is a vector of complex-valued elements which result from the cross-correlations of the desired received signal, say $s_1(t) * c_1(t)$, with the corresponding spreading sequence at the L_g delays, b is the desired symbol to be detected, and $\boldsymbol{z}$ represents the vector of additive Gaussian noise plus interference resulting from the cross-correlations of the spreading sequence with the received signals of the other users and intersymbol interference due to channel multipath. For a sufficiently large number of users and channel multipath components, the vector z may be characterized as complex-valued Gaussian with zero mean and covariance matrix $\boldsymbol{R}_z = E[zz^H]$. Based on this statistical characterization of z, the RAKE finger weight vector for maximum-likelihood detection is given as

$$\boldsymbol{w} = \boldsymbol{R}_z^{-1}\boldsymbol{g} \tag{13.5–36}$$

Given the channel impulse response, the implementation of the maximum-likelihood detector requires the evaluation of the covariance matrix $\boldsymbol{R}_z$ and the desired signal vector $\boldsymbol{g}$. The procedure for evaluation of these parameters has been described in a paper by Bottomley et al. (2000). Also investigated in this paper is the selection of the number of RAKE fingers and the selection of the corresponding delays for different channel characteristics.

In the description of the generalized RAKE demodulator given above, we assumed that the channel is time-invariant. In a randomly time-variant channel, the position of the RAKE fingers and the weights $\{w_i\}$ must be varied according to the characteristics of the channel impulse response. The pilot signal transmitted by the base station to the mobile receivers is used to estimate the channel impulse response, from which the finger placement and weights $\{w_i\}$ can be determined adaptively. The interested reader is referred to the paper by Bottomley et al. (2000) for a detailed description of the performance of the generalized RAKE demodulator for some channel models.

13.5–4 Receiver Structures for Channels with Intersymbol Interference

As described above, the wideband signal waveforms that are transmitted through the multipath channels resolve the multipath components with a time resolution of $1/W$, where W is the signal bandwidth. Usually, such wideband signals are generated as direct sequence spread spectrum signals, in which the PN spreading sequences are the outputs of linear feedback shift registers, e.g., maximum-length linear feedback shift registers. The modulation impressed on the sequences may be binary PSK, QPSK, DPSK, or binary orthogonal. The desired bit rate determines the bit interval or symbol interval.

The RAKE demodulator that we described above is the optimum demodulator based on the condition that the bit interval $T_b \gg T_m$, i.e., there is negligible ISI. When this condition is not satisfied, the RAKE demodulator output is corrupted by ISI. In such a case, an equalizer is required to suppress the ISI.

To be specific, we assume that binary PSK modulation is used and spread by a PN sequence. The bandwidth of the transmitted signal is sufficiently broad to resolve two or more multipath components. At the receiver, after the signal is demodulated to baseband, it may be processed by the RAKE, which is the matched filter to the channel response, followed by an equalizer to suppress the ISI. The RAKE output is sampled at the bit rate, and these samples are passed to the equalizer. An appropriate equalizer, in this case, would be a maximum-likelihood sequence estimator implemented by use

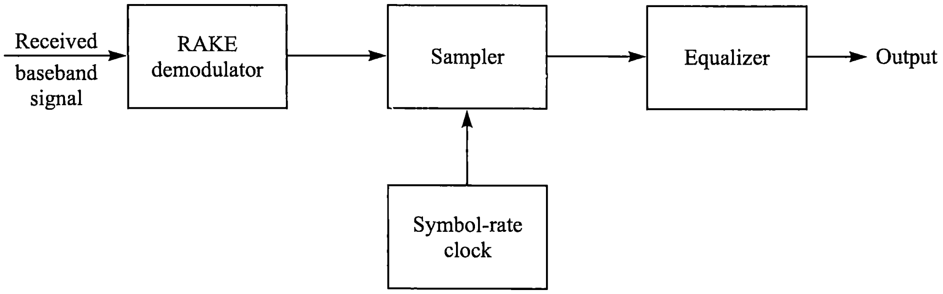

Received baseband signal
RAKE demodulator
Sampler
Equalizer
Output
Symbol-rate clock

FIGURE 13.5–10
Receiver structure for processing wideband signal corrupted by ISI.

of the Viterbi algorithm or a decision feedback equalizer (DFE). This demodulator structure is shown in Figure 13.5–10.

Other receiver structures are also possible. If the period of the PN sequence is equal to the bit interval, i.e., $LT_c = T_b$, where T_c is the chip interval and L is the number of chips per bit, a fixed filter matched to the spreading sequence may be used to process the received signal and followed by an adaptive equalizer, such as a fractionally spaced DFE, as shown in Figure 13.5–11. In this case, the matched filter output is sampled at some multiple of the chip rate, e.g., twice the chip rate, and fed to the fractionally spaced DFE. The feedback filter in the DFE would have taps spaced at the bit interval. The adaptive DFE would require a training sequence for adjustment of its coefficients to the channel multipath structure.

An even simpler receiver structure is one in which the spread spectrum matched filter is replaced by a low-pass filter whose bandwidth is matched to the transmitted signal bandwidth. The output of such a filter may be sampled at an integer multiple of the chip rate and the samples are passed to an adaptive fractionally spaced DFE. In this case, the coefficients of the feedback filter in the DFE, with the aid of a training sequence, will adapt to the combination of the spreading sequence and the channel multipath. Abdulrahman et al. (1994) consider the use of a DFE to suppress ISI in a CDMA system in which each user employs a wideband direct sequence spread spectrum signal.

The paper by Taylor et al. (1998) provides a broad survey of equalization techniques and their performance for wireless channels.

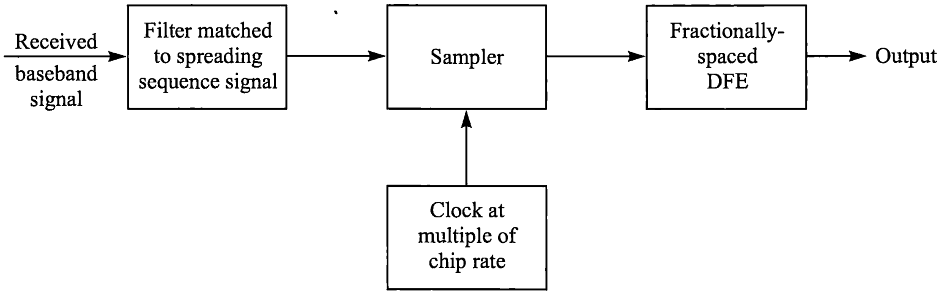

Received baseband signal
Filter matched to spreading sequence signal
Sampler
Fractionally-spaced DFE
Output
Clock at multiple of chip rate

FIGURE 13.5–11
Alternative receiver structure for processing wideband signal corrupted by ISI.

13.6
MULTICARRIER MODULATION (OFDM)

Multicarrier modulation was introduced in Chapter 11 (Section 11.2), and a special form of multicarrier transmission, called orthogonal frequency-division multiplexing (OFDM), was treated in detail. In this section, we consider the use of OFDM for digital transmission on fading multipath channels.

From our previous discussion, we have observed that OFDM is an attractive alternative to single-carrier modulation for use in time-dispersive channels. By selecting the symbol duration in an OFDM system to be significantly larger than the channel dispersion, intersymbol interference (ISI) can be rendered negligible and completely eliminated by use of a time guard band or, equivalently, by the use of a cyclic prefix embedded in the OFDM signal. The elimination of ISI due to multipath dispersion, without the use of complex equalizers, is a basic motivation for use of OFDM for digital communication in fading multipath channels. However, OFDM is especially vulnerable to Doppler spread resulting from time variations in the channel impulse response, as is the case in mobile communication systems. The Doppler spreading destroys the orthogonality of the OFDM subcarriers and results in intercarrier interference (ICI) which can severely degrade the performance of the OFDM system. In the following section we evaluate the effect of a Doppler spread on the performance of OFDM.

13.6–1 Performance Degradation of an OFDM System due to Doppler Spreading

Let us consider an OFDM system with N subcarriers $\{e^{j2\pi f_k t}\}$, where each subcarrier employs either M-ary QAM or PSK modulation. The subcarriers are orthogonal over the symbol duration T, i.e., $f_k = k/T$, $k = 1, 2, \ldots, N$, so that

$$\frac{1}{T}\int_0^T e^{j2\pi f_i t}\, e^{-j2\pi f_k t}\, dt = \begin{cases} 1 & k = i \\ 0 & k \neq i \end{cases} \tag{13.6–1}$$

The channel is modeled as a frequency-selective randomly varying channel with impulse response $c(\tau; t)$. Within the frequency band of each subcarrier, the channel is modeled as a frequency-nonselective Rayleigh fading channel with impulse response.

$$c_k(\tau; t) = \alpha_k(t)\delta(t), \qquad k = 0, 1, \ldots, N-1 \tag{13.6–2}$$

It is assumed that the processes $\{\alpha_k(t), k = 0, 1, \ldots, N-1\}$ are complex-valued, jointly stationary, and jointly Gaussian with zero means and cross-covariance function

$$R_{\alpha_k\alpha_i}(\tau) = E[\alpha_k(t+\tau)\alpha_t^*(t)], \qquad k, i = 0, 1, \ldots, N-1 \tag{13.6–3}$$

For each fixed k, the real and imaginary parts of the process $\alpha_k(t)$ are assumed independent with identical covariance function. It is further assumed that the covariance function $R_{\alpha_k\alpha_i}(\tau)$ has the following factorable form

$$R_{\alpha_k\alpha_i}(\tau) = R_1(\tau)R_2(k-i) \tag{13.6–4}$$

which is sufficient to represent the frequency selectivity and the time-varying effects of the channel. $R_1(\tau)$ represents the temporal correlation of the process $\alpha_k(t)$, which is identical for all $k = 0, 1, \ldots, N-1$, and $R_2(k)$ represents the correlation in frequency across subcarriers.

To obtain numerical results, we assume that the power spectral density corresponding to $R_1(\tau)$ is modeled as in Jakes (1974) and given by (see Figure 13.1–8)

$$\mathcal{S}(f) = \begin{cases} \dfrac{1}{\pi f_m \sqrt{1-(f/f_m)^2}} & |f| \le f_m \\ 0 & \text{otherwise} \end{cases} \tag{13.6–5}$$

where F_d is the maximum Doppler frequency. We note that

$$R_1(\tau) = J_0(2\pi f_m \tau) \tag{13.6–6}$$

where $J_0(\tau)$ is the zero-order Bessel function of the first kind. To specify the correlation in frequency across the subcarriers, we model the multipath power intensity profile as an exponential of the form

$$R_c(\tau) = \beta e^{-\beta\tau}, \qquad \tau > 0, \ \ \beta > 0 \tag{13.6–7}$$

where β is a parameter that controls the coherence bandwidth of the channel. The Fourier transform of $R_c(\tau)$ yields

$$R_C(f) = \frac{\beta}{\beta + j2\pi f} \tag{13.6–8}$$

which provides a measure of the correlation of the fading across the subcarriers, as shown in Figure 13.6–1. Hence, $R_2(k) = R_C(k/T)$ is the frequency separation between two adjacent subcarriers. The 3-dB bandwidth of $R_C(f)$ may be defined as the coherence bandwidth of the channel and is easily shown to be $\sqrt{3}\beta/2\pi$.

The channel model described above is suitable for modeling OFDM signal transmission in mobile radio systems, such as cellular systems and radio broadcasting systems. Since the symbol duration T is usually selected to be much larger than the channel multipath spread, it is reasonable to model the signal fading as flat over each subcarrier. However, compared with the entire OFDM system bandwidth W, the coherence bandwidth of the channel is usually smaller. Hence, the channel is frequency-selective over the entire OFDM signal bandwidth.

Let us now model the time variations of the channel within an OFDM symbol interval T. For mobile radio channels of practical interest, the channel coherence time is significantly larger than T. For such slow fading channels, we may use the two-term Taylor series expansion, first introduced by Bello (1963), to represent the time-varying channel variations $\alpha_k(t)$ as

$$\alpha_k(t) = \alpha_k(t_0) + \alpha_k'(t_0)(t - t_0), \qquad t_0 = \frac{T}{2}, \ \ 0 \le t \le T \tag{13.6–9}$$

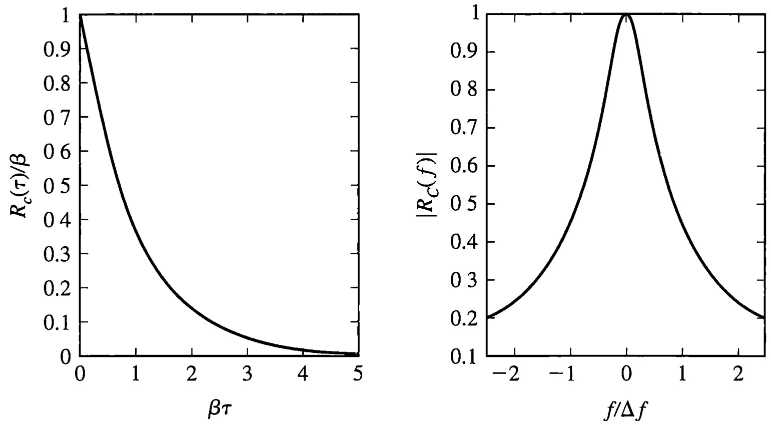


FIGURE 13.6–1
Multipath delay profile and frequency correlation function.

Therefore, the impulse response of the kth subchannel within a symbol interval is given as

$$c_k(\tau; t) = \alpha_k(t_0)\delta(\tau) + (t - t_0)\alpha_k'(t_0)\delta(\tau) \tag{13.6–10}$$

Since $R_1(\tau)$ given by Equation 13.6–6 is infinitely differentiable, all mean-square derivatives exist and hence the differentiation of $\alpha_k(t)$ is justified.

Based on the channel model described above, we determine the ICI term at the detector and evaluate its power. The baseband signal transmitted over the channel is expressed as

$$s(t) = \frac{1}{\sqrt{T}} \sum_{k=0}^{N-1} s_k\, e^{j2\pi f_k t}, \qquad 0 \le t \le T \tag{13.6–11}$$

where $f_k = k/T$ and s_k, $k = 0, 1, \ldots, N-1$, represents the complex-valued signal constellation points. We assume that

$$E\left[|s_k|^2\right] = 2\mathcal{E}_{\text{avg}} \tag{13.6–12}$$

where $2\mathcal{E}_{\text{avg}}$ denotes the average symbol energy of each s_k.

The received baseband signal may be expressed as

$$r(t) = \frac{1}{\sqrt{T}} \sum_{k=0}^{N-1} \alpha_k(t) s_k\, e^{j2\pi f_k t} + n(t) \tag{13.6–13}$$

where $n(t)$ is the additive noise, which is modeled as a complex-valued, zero-mean Gaussian process that is spectrally flat within the signal bandwidth with spectral density $2N_0$ W/Hz. By using the two-term Taylor series expansion for $a_k(t)$, $r(t)$ may be expressed as

$$r(t) = \frac{1}{\sqrt{T}} \sum_{k=0}^{N-1} \alpha_k\,(t_0) s_k e^{j2\pi f_k t} + \frac{1}{\sqrt{T}} \sum_{k=0}^{N-1} (t - t_0)\alpha_k'(t_0) s_k\, e^{j2\pi f_k t} + n(t) \tag{13.6–14}$$

The received signal in a symbol interval is passed through a parallel bank of N correlators, where each correlator is tuned to one of the N subcarrier frequencies. The output of the ith correlator at the sampling instant is

$$\begin{aligned}\hat{s}_i &= \frac{1}{\sqrt{T}} \int_0^T r(t)\, e^{-j2\pi f_i t}\, dt \\ &= \alpha_i(t_0)s_i + \frac{T}{2\pi j} \sum_{\substack{k=0 \\ k\neq i}}^{N-1} \frac{\alpha'_k(t_0)s_k}{k-i} + n_i\end{aligned} \tag{13.6–15}$$

The first term in Equation 13.6–15 represents the desired signal, the second term represents the ICI, and the third term is the additive noise component.

The mean-square value of the desired signal component is

$$\begin{aligned}S &= E\left[|\alpha_i(t_0)s_i|^2\right] \\ &= E\left[|\alpha_i(t_0)|^2\right] E\left[|s_i|^2\right] = 2\mathcal{E}_{\text{avg}}\end{aligned} \tag{13.6–16}$$

where the average channel gain is normalized to unity. The mean-square value of the ICI term is evaluated as follows. Since $R_{\alpha_s a_k}(\tau) = R_1(\tau)$ is infinitely differentiable, all (mean-square) derivatives of the process $\alpha_k(t)$, $-\infty < t < \infty$, exist. In particular, the first derivative $\alpha'_k(t)$ is a zero-mean, complex-valued Gaussian process with correlation function

$$E\left[\alpha'_k(t+\tau)(\alpha'_k(t)^*)\right] = -R''_1(\tau) \tag{13.6–17}$$

with corresponding spectral density $(2\pi f)^2 \mathcal{S}(f)$. Hence,

$$E\left[|\alpha'_k(t)|^2\right] = \int_{-f_m}^{f_m} (2\pi f)^2 \mathcal{S}(f)\, df = 2\pi^2 f_m^2 \tag{13.6–18}$$

The power in the ICI term is

$$\begin{aligned}I &= E\left[\left|\frac{T}{2\pi j} \sum_{\substack{k=0 \\ k\neq i}}^{N-1} \frac{a'_k(t_0)s_k}{k-i}\right|^2\right] \\ &= \left(\frac{T}{2\pi}\right)^2 \sum_{\substack{k=0 \\ k\neq i}}^{N-1} \sum_{\substack{l=0 \\ l\neq i}}^{N-1} \frac{1}{(k-i)(l-i)} E\left[\alpha'_k(t_0)s_k\,(\alpha'_l(t_0)s_l)^*\right] \\ &+ \left(\frac{T}{2\pi}\right)^2 \sum_{\substack{k=0 \\ k\neq i}}^{N-1} \frac{1}{(k-i)^2} E\left[|\alpha'_k(t_0)s_k|^2\right]\end{aligned} \tag{13.6–19}$$

We note that the pair $(\alpha'_k(t_0), \alpha'_l(t_0))$ is statistically independent of (s_k, s_l). Furthermore, the $\{s_k\}$ are iid with zero means. Hence, the first term of the right-hand side of

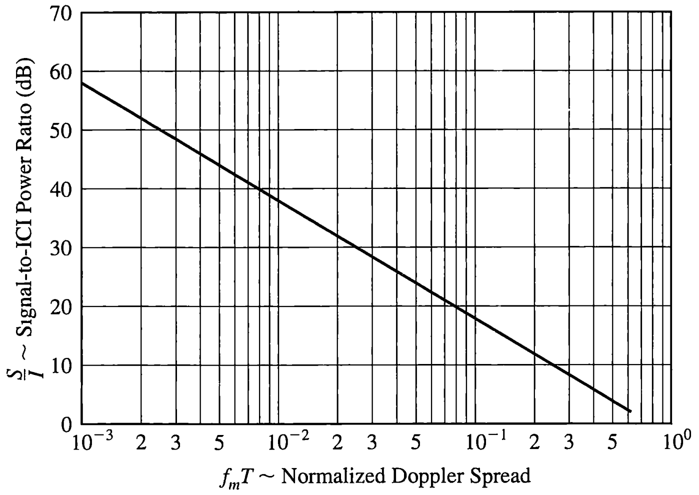


FIGURE 13.6–2
Signal-to-ICI power ratio versus normalized Doppler spread.

Equation 13.6–19 is zero. Therefore, by using the result from Equation 13.6–18 in Equation 13.6–19, the power of the ICI component is

$$I = \frac{\left(Tf_m\right)^2}{2} \sum_{\substack{k=0 \\ k \neq i}}^{N-1} \frac{2\mathcal{E}_s}{(k-i)^2} \tag{13.6–20}$$

Consequently, the signal-to-interference ratio S/I is given by

$$\frac{S}{I} = \frac{1}{\dfrac{(Tf_m)^2}{2} \displaystyle\sum_{\substack{k=0 \\ k \neq 1}}^{N-1} \frac{1}{(k-i)^2}} \tag{13.6–21}$$

Graphs of S/I versus Tf_m are shown in Figure 13.6–2 for $N = 256$ subcarriers and $i = N/2$, the interference on the middle subcarrier.

The evaluation of the effect of the ICI on the error rate performance of an OFDM system requires knowledge of the PDF of the ICI which, in general, is a mixture of Gaussian PDFs. However, when the number of subcarriers is large, the distribution of the ICI can be approximated by a Gaussian distribution, and thus the evaluation of the error rate performance is straightforward.

Figure 13.6–3 illustrates the symbol error probability for an OFDM system having $N = 256$ subcarriers and 16-QAM, where the error probability is evaluated analytically based on the Gaussian model for the ICI and by Monte Carlo simulation. We observe that the ICI severely degrades the performance of the OFDM system. In the following section we describe a method for suppressing the ICI and, thus, improving the performance of the OFDM system.

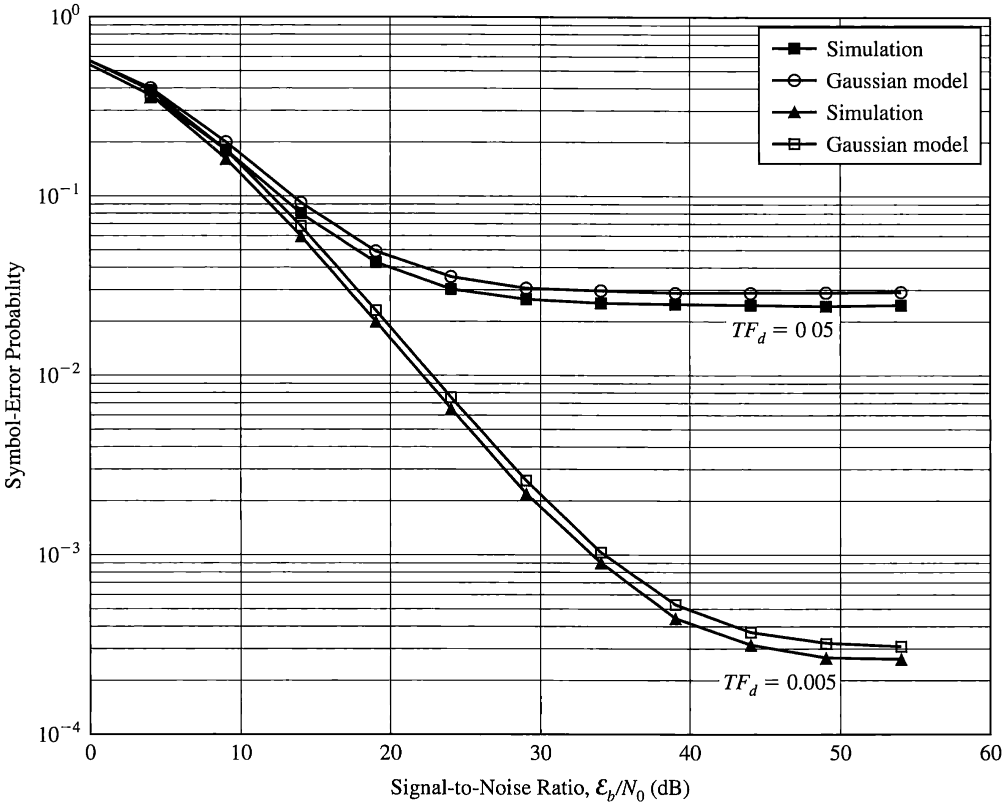


FIGURE 13.6–3

Symbol error probability for 16-QAM OFDM system with $N = 256$ subcarriers.

13.6–2 Suppression of ICI in OFDM Systems

The distortion caused by ICI in an OFDM system is akin to the distortion caused by ISI in a single-carrier system. Recall that a linear time-domain equalizer based on the minimum mean-square-error (MMSE) criterion is an effective method for suppressing ISI. In a similar manner, we may apply the MMSE criterion to suppress the ICI in the frequency domain. Thus, we begin with the N frequency samples at the output of the discrete Fourier transform (DFT) processor, which we denote by the vector $\boldsymbol{R}(m)$ for the mth frame. Then we form the estimate of the symbol $s_k(m)$ as

$$\hat{s}_k(m) = \boldsymbol{b}_k^H(m)\boldsymbol{R}(m), \qquad k = 0, 1, \ldots, N-1 \tag{13.6–22}$$

where $\boldsymbol{b}_k(m)$ is the coefficient vector of size $N \times 1$. This vector is selected to minimize the MSE

$$E\left[|s_k(m) - \hat{s}_k(m)|^2\right] = E\left[|s_k(m) - \boldsymbol{b}_k^H(m)\boldsymbol{R}(m)|^2\right] \tag{13.6–23}$$

where the expectation is taken with respect to the signal and noise statistics. By applying the orthogonality principle, the optimum coefficient vector is obtained as

$$\boldsymbol{b}_k(m) = \left[\boldsymbol{G}(m)\boldsymbol{G}^H(m) + \sigma^2\boldsymbol{I}_N\right]^{-1}\boldsymbol{g}_k(m), \qquad k = 0, 1, \ldots, N-1 \tag{13.6–24}$$

where

$$\begin{aligned} E\left[\boldsymbol{R}(m)\boldsymbol{R}^H(m)\right] &= \boldsymbol{G}(m)\boldsymbol{G}^H(m) + \sigma^2\boldsymbol{I}_N \\ E\left[\boldsymbol{R}(m)s_k^H(m)\right] &= \boldsymbol{g}_k(m) \end{aligned} \tag{13.6–25}$$

and $\boldsymbol{G}(m)$ is related to the channel impulse response matrix $\boldsymbol{H}(m)$ through the DFT relation (see Problem 13.16)

$$\boldsymbol{G}(m) = \boldsymbol{W}^H\boldsymbol{H}(m)\boldsymbol{W} \tag{13.6–26}$$

where $\boldsymbol{W}$ is the orthonormal (IDFT) transformation matrix. The vector $\boldsymbol{g}_k(m)$ is the kth column of the matrix $\boldsymbol{G}(m)$, and σ^2 is the variance of the additive noise component. It is easily shown that the minimum MSE for the signal on the kth subcarrier may be expressed as

$$E\left[|s_k(m) - \hat{s}_k(m)|^2\right] = 1 - \boldsymbol{g}_k^H(m)(\boldsymbol{G}(m)\boldsymbol{G}^H(m) + \sigma^2\boldsymbol{I}_N)^{-1}\boldsymbol{g}_k(m) \tag{13.6–27}$$

We observe that the optimum weight vectors $\{\boldsymbol{b}_k(m)\}$ require knowledge of the channel impulse response. In practice, the channel response may be estimated by periodically transmitting pilot signals on each of the subcarriers and by employing a decision-directed method when data are transmitted on the N subcarriers. In a slowly fading channel, the coefficient vectors $\{\boldsymbol{b}_k(m)\}$ may also be adjusted recursively by employing either an LMS- or an RLS-type algorithm, as previously described in the context of equalization for suppression of ISI.

13.7
BIBLIOGRAPHICAL NOTES AND REFERENCES

In this chapter, we have considered a number of topics concerned with digital communications over a fading multipath channel. We began with a statistical characterization of the channel and then described the ramifications of the channel characteristics on the design of digital signals and on their performance. We observed that the reliability of the communication system is enhanced by the use of diversity transmission and reception. We also considered the transmission of digital information through time-dispersive channels and described the RAKE demodulator, which is the matched filter for the channel. Finally, we considered the use of OFDM for mobile communications and on the performance of an OFDM system, described the effect of ICI caused by Doppler frequency spreading.

The pionerring work on the characterization of fading multipath channels and on signal and receiver design for reliable digital communciations over such channels was done by Price (1954, 1956). This work was followed by additional significant contributions from Price and Green (1958, 1960), Kailath (1960, 1961), and Green (1962). Diversity transmission and diversity combining techniques under a variety of channel conditions have been considered in the papers by Pierce (1958), Brennan (1959), Turin (1961, 1962), Pierce and Stein (1960), Barrow (1963), Bello and Nelin (1962a, b, 1963), Price (1962a, b), and Lindsey (1964).

Our treatment of digital communications over fading channels focused primarily on the Rayleigh fading channel model. For the most part, this is due to the wide acceptance of this model for describing the fading effects on many radio channels and to its mathematical tractability. Although other statistical models, such as the Ricean fading model or the Nakagami fading model may be more appropriate for characterizing fading on some real channels, the general approach in the design of reliable communications presented in this chapter carries over. Alouini and Goldsmith (1998), Simon and Alouini (1988, 2000), and Annamalai et al. (1998, 1999) have presented a unified approach to evaluating the error rate performance of digital communication systems for various fading channel models. The effect of ICI in OFDM for mobile communications has been extensively treated in the literature, e.g., the papers by Robertson and Kaiser (1999), Li and Kavehrad (1999), Ciavaccini and Vitetta (2000), Li and Cimini (2001), Stamoulis et al. (2002), and Wang et al. (2006). A general treatment of wireless communications is given in the books by Rappaport (1996) and Stuber (2000).

PROBLEMS

13.1 The scattering function $S(\tau; \lambda)$ for a fading multipath channel is nonzero for the range of values $0 \leq \tau \leq 1$ ms and -0.1 Hz $\leq \lambda \leq 0.1$ Hz. Assume that the scattering function is approximately uniform in the two variables.

a. Give numerical values for the following parameters:
  (i) The multipath spread of the channel.
  (ii) The Doppler spread of the channel.
  (iii) The coherence time of the channel.
  (iv) The coherence bandwidth of the channel.
  (v) The spread factor of the channel.

b. Explain the meaning of the following, taking into consideration the answers given in (a):
  (i) The channel is frequency-nonselective.
  (ii) The channel is slowly fading.
  (iii) The channel is frequency-selective.

c. Suppose that we have a frequency allocation (bandwidth) of 10 kHz and we wish to transmit at a rate of 100 bits over this channel. Design a binary communication system with frequency diversity. In particular, specify
  (i) The type of modulation.
  (ii) The number of subchannels.
  (iii) The frequency separation between adjacent carriers.
  (iv) The signaling interval used in your design.

Justify your choice of parameters.

13.2 Consider a binary communication system for transmitting a binary sequence over a fading channel. The modulation is orthogonal FSK with third-order frequency diversity ($L = 3$). The demodulator consists of matched filters followed by square-law detectors. Assume that the FSK carriers fade independently and identically according to a Rayleigh envelope distribution. The additive noises on the diversity signals are zero-mean Gaussian with autocorrelation functions $E[z_k^*(t)z_k(t+\tau)] = 2N_0\delta(\tau)$. The noise processes are mutually statistically independent.

a. The transmitted signal may be viewed as binary FSK with square-law detection, generated by a repetition code of the form

$$1 \rightarrow \boldsymbol{c}_1 = [1 \quad 1 \quad 1], \qquad 0 \rightarrow \boldsymbol{c}_0 = [0 \quad 0 \quad 0]$$

Determine the error rate performance P_{bh} for a hard-decision decoder following the square-law-detected signals.

b. Evaluate P_{bh} for $\overline{\gamma}_c = 100$ and 1000.

c. Evaluate the error rate P_{bs} for $\overline{\gamma}_c = 100$ and 1000 if the decoder employs soft-decision decoding.

d. Consider the generalization of the result in (a). If a repetition code of block length L (L odd) is used, determine the error probability P_{bh} of the hard-decision decoder and compare that with P_{bs}, the error rate of the soft-decision decoder. Assume $\overline{\gamma} \gg 1$.

13.3 Suppose that the binary signal $\pm s_l(t)$ is transmitted over a fading channel and the received signal is

$$r_l(t) = \pm a s_l(t) + z(t), \qquad 0 \leq t \leq T$$

where $z(t)$ is zero-mean white Gaussian noise with autocorrelation function

$$R_{zz}(\tau) = 2N_0\delta(\tau)$$

The energy in the transmitted signal is $\mathcal{E} = \frac{1}{2}\int_0^T |s_l(t)|^2\, dt$. The channel gain a is specified by the probability density function

$$p(a) = 0.1\delta(a) + 0.9\delta(a-2)$$

a. Determine the average probability of error P_b for the demodulator that employs a filter matched to $s_l(t)$.

b. What value does P_b approach as $\mathcal{E}/N_0$ approaches infinity?

c. Suppose that the same signal is transmitted on two statistically *independently fading* channels with gains a_1 and a_2, where

$$p(a_k) = 0.1\delta(a_k) + 0.9\delta(a_k - 2), \qquad k = 1, 2$$

The noises on the two channels are statistically independent and identically distributed. The demodulator employs a matched filter for each channel and simply adds the two filter outputs to form the decision variable. Determine the average P_b.

d. For the case in (c) what value does P_b approach as $\mathcal{E}/N_0$ approaches infinity?

13.4 A multipath fading channel has a multipath spread of $T_m = 1$ s and a Doppler spread $B_d = 0.01$ Hz. The total channel bandwidth at bandpass available for signal transmission is $W = 5$ Hz. To reduce the effects of intersymbol interference, the signal designer selects a pulse duration $T = 10$ s.

a. Determine the coherence bandwidth and the coherence time.

b. Is the channel frequency selective? Explain.

c. Is the channel fading slowly or rapidly? Explain.

d. Suppose that the channel is used to transmit binary data via (antipodal) coherently detected PSK in a frequency diversity mode. Explain how you would use the available channel bandwidth to obtain frequency diversity and determine how much diversity is available.

e. For the case in (d), what is the *approximate* SNR required per diversity to achieve an error probability of 10^{-6}?

f. Suppose that a wideband signal is used for transmission and a RAKE-type receiver is used for demodulation. How many taps would you use in the RAKE receiver?

g. Explain whether or not the RAKE receiver can be implemented as a coherent receiver with maximal ratio combining.

h. If binary orthogonal signals are used for the wideband signal with square-law post-detection combining in the RAKE receiver, what is the *approximate* SNR required to achieve an error probability of 10^{-6}? (Assume that all taps have the same SNR.)

13.5 In the binary communication system shown in Figure P13.5, $z_1(t)$ and $z_2(t)$ are statistically independent white Gaussian noise processes with zero-mean and identical autocorrelation functions $R_{zz}(\tau) = 2N_0\delta(\tau)$. The sampled values U_1 and U_2 represent the *real parts* of the matched filter outputs. For example, if $s_l(t)$ is transmitted, then we have

$$U_1 = 2\mathcal{E} + N_1$$
$$U_2 = N_1 + N_2$$

where $\mathcal{E}$ is the transmitted signal energy and

$$N_k = \operatorname{Re}\left[\int_0^T s_l^*(t) z_k(t)\, dt\right], \qquad k = 1, 2$$

It is apparent that U_1 and U_2 are correlated Gaussian variables while N_1 and N_2 are independent Gaussian variables. Thus,

$$p(n_1) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{n_1^2}{2\sigma^2}\right)$$
$$p(n_2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{n_2^2}{2\sigma^2}\right)$$

where the variance of N_k is $\sigma^2 = 2\mathcal{E}N_0$.

a. Show that the joint probability density function for U_1 and U_2 is

$$p(u_1, u_2) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{1}{\sigma^2}\left[(u_2 - 2\mathcal{E})^2 - u_2(u_1 - 2\mathcal{E}) + \tfrac{1}{2}u_2^2\right]\right\}$$

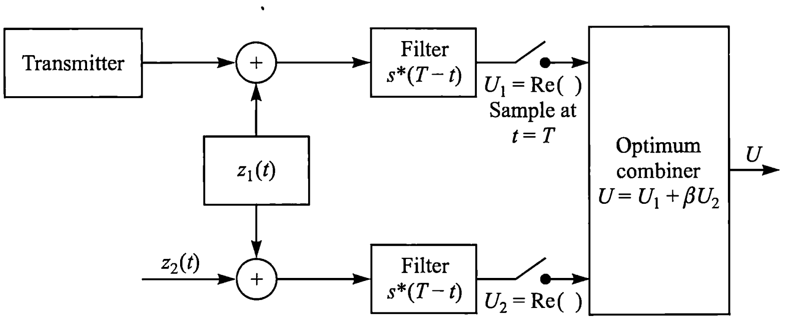


FIGURE P13.5

if $s(t)$ is transmitted and

$$p(u_1, u_2) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{1}{\sigma^2}\left[(u_1 + 2\mathcal{E})^2 - u_2(u_1 + 2\mathcal{E}) + \tfrac{1}{2}u_2^2\right]\right\}$$

if $-s(t)$ is transmitted.

b. Based on the likelihood ratio, show that the optimum combination of U_1 and U_2 results in the decision variable

$$U = U_1 + \beta U_2$$

where β is a constant. What is the optimum value of β?

c. Suppose that $s(t)$ is transmitted. What is the probability density function of U?

d. What is the probability of error assuming that $s(t)$ was transmitted? Express your answer as a function for the SNR $\mathcal{E}/N_0$.

e. What is the loss in performance if only $U = U_1$ is the decision variable?

13.6 Consider the model for a binary communication system with diversity as shown in Figure P13.6. The channels have fixed attenuations and phase shifts. The $\{z_k(t)\}$ are complex-valued white Gaussian noise processes with zero-mean and autocorrelation functions

$$R_{zz}(t) = E\left[z_k^*(t)z_k(t+\tau)\right] = 2N_{0k}\delta(\tau)$$

(Note that the spectral densities $\{N_{0k}\}$ are all different.) Also, the noise processes $\{z_k(t)\}$ are mutually statistically independent. The $\{\beta_k\}$ are complex-valued weighting factors to be determined. The decision variable from the combiner is

$$U = \mathrm{Re}\left(\sum_{k=1}^{L} \beta_k U_k\right) \underset{-1}{\overset{1}{\gtrless}} 0$$

a. Determine the PDF $p(u)$ when $+1$ is transmitted.

b. Determine the probability of error P_b as a function of the weights $\{\beta_k\}$.

c. Determine the values of $\{\beta_k\}$ that minimize P_b.

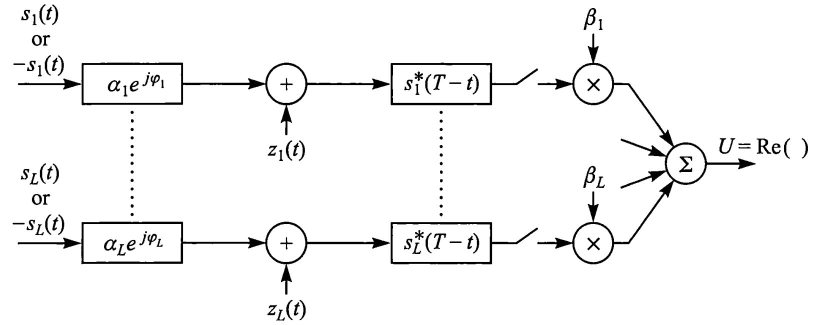


FIGURE P13.6

13.7 Determine the probability of error for binary orthogonal signaling with Lth-order diversity over a Rayleigh fading channel. The PDFs of the two decision variables are given by Equations 13.4–31 and 13.4–32.

13.8 A binary sequence is transmitted via binary antipodal signaling over a Rayleigh fading channel with Lth-order diversity. When $s_l(t)$ is transmitted, the received equivalent lowpass signals are

$$r_k(t) = \alpha_k e^{j\phi_k} s_l(t) + z_k(t), \qquad k = 1, 2, \ldots, L$$

The fading among the L subchannels is statistically independent. The additive noise terms $\{z_k(t)\}$ are zero-mean, statistically independent, and identically distributed white Gaussian noise processes with autocorrelation function $R_{zz}(\tau) = 2N_0\delta(\tau)$. Each of the L signals is passed through a filter matched to $s_l(t)$ and the output is phase-corrected to yield

$$U_k = \text{Re}\left[e^{-j\phi_k} \int_0^T r_k(t) s_l^*(t)\, dt\right], \qquad k = 1, 2, \ldots, L$$

The $\{U_k\}$ are combined by a linear combiner to form the decision variable

$$U = \sum_{k=1}^{L} U_k$$

a. Determine the PDF of U conditional on fixed values for the $\{\alpha_k\}$.
b. Determine the expression for the probability of error when the $\{\alpha_k\}$ are statistically independent and identically distributed Rayleigh random variables.

13.9 The Chernov bound for the probability of error for binary FSK with diversity L in Rayleigh fading was shown to be

$$P_2(L) < [4p(1-p)]^L = \left[4\frac{1+\bar{\gamma}_c}{(2+\bar{\gamma}_c)^2}\right]^L$$
$$< 2^{-\bar{\gamma}_b g(\bar{\gamma}_c)}$$

where

$$g(\bar{\gamma}_c) = \frac{1}{\bar{\gamma}_c} \log_2 \left[\frac{(2+\bar{\gamma}_c)^2}{4(1+\bar{\gamma}_c)}\right]$$

a. Plot $g(\bar{\gamma}_c)$ and determine its approximate maximum value and the value of $\bar{\gamma}_c$ where the maximum occurs.
b. For a given $\bar{\gamma}_b$, determine the optimal order of diversity.
c. Compare $P_2(L)$, under the condition that $g(\bar{\gamma}_c)$ is maximized (optimal diversity), with the error probability for binary FSK and AWGN with no fading, which is

$$P_2 = \tfrac{1}{2} e^{-\gamma_b/2}$$

and determine the penalty in SNR due to fading and noncoherent (square-law) combining.

13.10 A DS spread spectrum system is used to resolve the multipath signal components in a two-path radio signal propagation scenario. If the path length of the secondary path is 300 m longer than that of the direct path, determine the minimum chip rate necessary to resolve the multipath components.

13.11 A baseband digital communication system employs the signals shown in Figure P13.11(a) for the transmission of two equiprobable messages. It is assumed that the communication problem studied here is a "one-shot" communication problem; that is, the above messages are transmitted just once and no transmission takes place afterward. The channel has no attenuation ($\alpha = 1$), and the noise is AWGN with power spectral density $\frac{1}{2}N_0$.

a. Find an appropriate orthonormal basis for the representation of the signals.

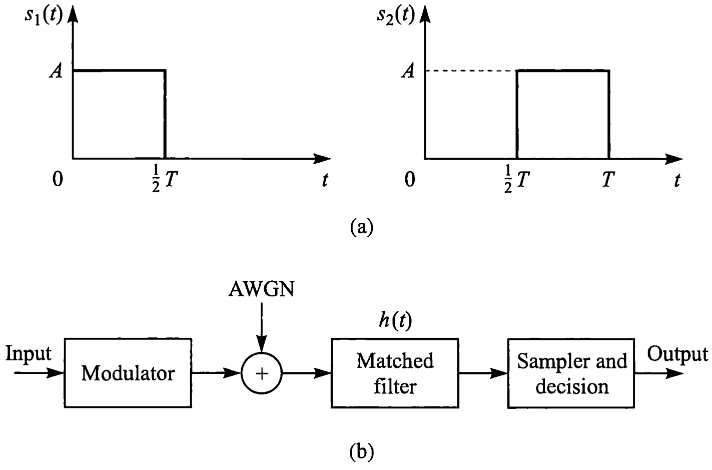
b. In a block diagram, give the precise specifications of the optimum receiver using matched filters. Label the diagram carefully.

c. Find the error probability of the optimum receiver.

d. Show that the optimum receiver can be implemented by using just *one* filter (see the block diagram in Figure P13.11(b)). What are the characteristics of the matched filter, the sampler and decision device?

e. Now assume that the channel is not ideal but has an impulse response of $c(t) = \delta(t) + \frac{1}{2}\delta(t - \frac{1}{2}T)$. Using the same matched filter as in (d), design the optimum receiver.

f. Assuming that the channel impulse response is $c(t) = \delta(t) + a\delta(t - \frac{1}{2}T)$, where a is a random variable uniformly distributed on [0, 1], and using the same matched filter as in (d), design the optimum receiver.



(a)

(b)

FIGURE P13.11

13.12 A communication system employs dual antenna diversity and binary orthogonal FSK modulation. The received signals at the two antennas are

$$r(t) = \alpha_1 s(t) + n_1(t)$$
$$r_2(t) = \alpha_2 s(t) + n_2(t)$$

where α_1 and α_2 are statistically iid Rayleigh random variables, and $n_1(t)$ and $n_2(t)$ are statistically independent, zero-mean and white Gaussian random processes with power-spectral density $\frac{1}{2}N_0$. The two signals are demodulated, squared, and then combined (summed) prior to detection.

a. Sketch the functional block diagram of the entire receiver, including the demodulator, the combiner, and the detector.

b. Plot the probability of error for the detector and compare the result with the case of no diversity.

13.13 The two equivalent lowpass signals shown in Figure P13.13 are used to transmit a binary sequence. The equivalent low-pass impulse response of the channel is $h(t) = 4\delta(t) - 2\delta(t-T)$. To avoid pulse overlap between successive transmissions, the transmission rate in bits/s is selected to be $R = 1/2T$. The transmitted signals are equally probable and are corrupted by additive zero-mean white Gaussian noise having an equivalent lowpass representation $z(t)$ with an autocorrelation function

$$R_{zz}(\tau) = E[z^*(t)z(t+\tau)] = 2N_0\delta(\tau)$$

a. Sketch the two possible equivalent lowpass noise-free *received* waveforms.

b. Specify the optimum receiver and sketch the equivalent lowpass impulse responses of all filters used in the optimum receiver. Assume *coherent detection* of the signals.

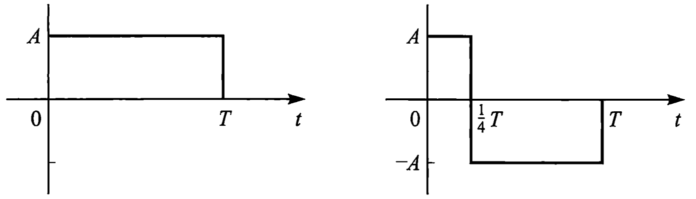


FIGURE P13.13

13.14 Verify the relation in Equation 13.3–14 by making the change of variable $\gamma = \alpha^2\mathcal{E}_b/N_0$ in the Nakagami-*m* distribution.

13.15 Consider a digital communication system that uses two transmitting antennas and one receiving antenna. The two transmitting antennas are sufficiently separated so as to provide dual spatial diversity in the transmission of the signal. The transmission scheme is as follows: If s_1 and s_2 represent a pair of symbols from either a one-dimensional or a two-dimensional signal constellation, which are to be transmitted by the two antennas, the signal from the first antenna over two signal intervals is (s_1, s_2^*) and from the second antenna the transmitted signal is $(s_2, -s_1^*)$. The signal received by the single receiving antenna over the two signal intervals is

$$\begin{aligned} r_1 &= h_1 s_1 + h_2 s_2 + n_1 \\ r_2 &= h_1 s_2^* - h_2 s_1^* + n_2 \end{aligned}$$

where (h_1, h_2) represent the complex-valued channel path gains, which may be assumed to be zero-mean, complex Gaussian with unit variance and statistically independent. The channel path gains (h_1, h_2) are assumed to be constant over the two signal intervals and known to the receiver. The terms (n_1, n_2) represent additive white Gaussian noise terms that have zero-mean and variance σ^2 and uncorrelated.

a. Show how to recover the transmitted symbols (s_1, s_2) from (r_1, r_2) and achieve dual diversity reception.

b. If the energy in the pair (s_1, s_2) is $(\mathcal{E}_s, \mathcal{E}_s)$ and the modulation is binary PSK, determine the probability of error.

c. Repeat (b) if the modulation is QPSK.

13.16 In the suppression of ICI in on DFDM system, the received signal vector for the *m*th frame may be expressed as

$$\boldsymbol{r}(m) = \boldsymbol{H}(m)\boldsymbol{W}\boldsymbol{s}(m) + \boldsymbol{n}(m)$$

where $\boldsymbol{W}$ is the $N \times N$ IDFT transformation matrix, $\boldsymbol{s}(m)$ is the $N \times 1$ signal vector, $\boldsymbol{n}(m)$ is the zero-mean, Gaussian noise vector with iid components, and $\boldsymbol{H}(m)$ is the $N \times N$ channel impulse response matrix, defined as

$$\boldsymbol{H}(m) = [\boldsymbol{h}^H(0, m)\, \boldsymbol{h}^H(1, m) \cdots \boldsymbol{h}^H(N-1, m)]^H$$

where $\boldsymbol{h}(n, m)$ is the right cyclic shift by $n + 1$ positions of the zero-padded channel impulse response vector of dimension $N \times 1$.

By expressing the DFT of $\boldsymbol{r}(m)$ by $\boldsymbol{R}(m)$, derive the relations in Equations 13.6–24, 13.6–25, and 13.6–27, where $\boldsymbol{G}(m)$ is defined in Equation 13.6–26.

13.17 Prove the result given in Equation 13.6–17.

13.18 Prove the result given in Equation 13.6–18.

14

Fading Channels II: Capacity and Coding

This chapter studies capacity and coding aspects for fading channels. In Chapter 13 the physical sources of the fading phenomenon in communications were discussed, and different models for fading channels were introduced. In particular, we saw that the effect of fading can be expressed in terms of the multipath spread of the channel denoted by T_m and the Doppler spread of the channel denoted by B_d. Equivalently we can use the coherence bandwidth and the coherence time of the channel denoted by $(\Delta f)_c$ and $(\Delta t)_c$, respectively. If two narrow pulses are separated by less than the coherence time of the channel, they will experience the same fading effects; and if two frequency tones are separated by less than the coherence bandwidth, they will be affected by the same fading effects. If the signal bandwidth is much larger than the coherence bandwidth of the channel, i.e., if $W \gg (\Delta f)_c$, then we have a frequency-selective channel model; and if $W \ll (\Delta f)_c$, then the channel model is frequency-nonselective or flat in frequency. In this case all frequency components of the input signal experience the same fading effects. Similarly if the signal duration is much longer than the channel coherence time, i.e., $T \gg (\Delta t)_c$, the signal will be subject to different fading effects and we have a fast fading channel; and if $T \ll (\Delta t)_c$ we have a slowly fading channel, or the channel is flat in time. Since the bandwidth and the duration of a signal are related through the approximate relation $W \approx 1/T$, we conclude that if in a channel $T_m B_d \ll 1$, i.e., if the channel is underspread, then we can choose a signal bandwidth W such that for this signal the channel is flat in both time and frequency.[†]

In dealing with capacity and coding for fading channels, we need to study channel variations during transmission of a block of signal waveforms transmitted over the channel. We can distinguish two different possibilities. In one case the characteristics of the channel change fast enough with respect to the transmission duration of a block that a single block of information experiences all possible realizations of the channel frequently. In this case the time averages during the transmission duration of a single block are equal to the statistical (ensemble) averages over all possible channel realizations. Another possibility is that the block duration is short and each block experiences only a cross section of channel characteristics. In this model, the channel remains relatively constant during the transmission of one block, and we can say that each block experiences a single state of the channel and the following blocks experience different channel states. The notions of channel capacity in these two cases are quite different. In the first channel model, since all channel realizations are experienced during a block, an ergodic channel model is appropriate and *ergodic capacity* can be defined as the ensemble average of channel capacity over all possible channel realizations. In the second channel model, where in each block different channel realizations are experienced, for each block the capacity will be different. Thus, the capacity can best be modeled as a random variable. In this case another notion of capacity known as *outage capacity* is more appropriate.

[†] We are excluding the spread spectrum systems in which $W \approx 1/T_c$ where T_c is the chip interval.

Another parameter that affects the capacity of fading channels is whether information about the state of the channel is available at the transmitter and/or the receiver. Availability of state information at the receiver that is usually measured by transmitting tones over the channel at different frequencies helps the receiver in increasing the channel capacity since the state of the channel can be interpreted as an auxiliary channel output. Availability of the state information at the transmitter makes it possible for the transmitter to design its signal to match the state of the channel through some kind of precoding. In this case the transmitter can change the level of the transmitted power according to the channel state, thus preserving transmission of valuable power during the time the channel is in deep fade and saving it for transmission during periods when the channel does not highly attenuate the transmitted signal.

Coding for fading channels introduces new challenges and opportunities that are different from the standard additive white Gaussian noise channels. As we will see in this chapter, the metrics that determine the performance of coding schemes over fading channels are different from the standard metrics used to compare the performance of different coding schemes over additive white Gaussian noise channels. On the other hand, since coding techniques introduce redundancy through transmission of the parity check codes, the extra transmissions provide diversity that improves the performance of coded systems over fading channels.

In this chapter we study the case of single-antenna systems from an information-theoretic and coding point of view. The study of capacity and coding for multiple-antenna systems and the design and analysis of space-time codes are done in Chapter 15.

14.1
CAPACITY OF FADING CHANNELS

The *capacity* of a channel is defined as the supremum of the rates at which reliable communication over the channel is possible. Reliable communication at rate R is possible if there exists a sequence of codes with rate R for which the average error probability tends to zero as the block length of the code increases. In other words, at any rate less than capacity we can find a code whose error probability is less than any specified $\epsilon > 0$. In Chapter 6 we gave a general expression for the capacity of a discrete memoryless channel in the form

$$C = \max_{p(x)} I(X; Y) \tag{14.1–1}$$

where the maximum is taken over all channel input probability density functions. For a power-constrained discrete-time AWGN channel, the capacity can be expressed as

$$C = \frac{1}{2} \log \left(1 + \frac{P}{N}\right) \tag{14.1–2}$$

where P is the signal power, N is the noise power, and C is the capacity in bits per transmission, or bits per (real) dimension. For a complex-input complex-output channel with circular complex Gaussian noise[†] with noise variance N_0, or $N_0/2$ per real and imaginary components, the capacity is given by

$$C = \log \left(1 + \frac{P}{N_0}\right) \tag{14.1–3}$$

bits per complex dimension.

The capacity of an ideal band-limited, power-limited additive white Gaussian waveform channel is given by

$$C = W \log \left(1 + \frac{P}{N_0 W}\right) \tag{14.1–4}$$

where W denotes the bandwidth, P denotes the signal power, and $N_0/2$ is the noise power spectral density. The capacity C in this case is given in bits per second. For an infinite-bandwidth channel in which the signal-to-noise ratio $P/(N_0 W)$ tends to zero, the capacity is given in Equation 6.5–44 as

$$C = \frac{1}{\ln 2} \frac{P}{N_0} \approx 1.44 \frac{P}{N_0} \tag{14.1–5}$$

The capacity in bits/sec/Hz (or bits per complex dimension) which determines the highest achievable spectral bit rate is given by

$$C = \log(1 + \text{SNR}) \tag{14.1–6}$$

where SNR denotes the signal-to-noise ratio defined as

$$\text{SNR} = \frac{P}{N_0 W} \tag{14.1–7}$$

Note that since $W \sim \frac{1}{T_s}$, where T_s is the symbol duration, the above expression for SNR can be written as $\text{SNR} = \frac{PT_s}{N_0} = \frac{\mathcal{E}_s}{N_0}$ where $\mathcal{E}_s$ indicates energy per symbol. In an AWGN channel the capacity is achieved by using a Gaussian input probability density function. At low values of SNR we have

$$C \approx \frac{1}{\ln 2} \text{SNR} \approx 1.44 \text{ SNR} \tag{14.1–8}$$

[†] We use the notation $\mathcal{CN}(0, \sigma^2)$ to denote a circular complex random variable with variance $\sigma^2/2$ per real and imaginary parts.

The notion of capacity for a band-limited additive white Gaussian noise channel can be extended to a nonideal channel in which the channel frequency response is denoted by $C(f)$. In this case the channel is described by the input-output relation of the form

$$y(t) = x(t) \star c(t) + n(t) \tag{14.1–9}$$

where $c(t)$ denotes the channel impulse response and $C(f) = \mathscr{F}[c(t)]$ is the channel frequency response. The noise is Gaussian with a power spectral density of $\mathcal{S}_n(f)$. It was shown in Chapter 11 that the capacity of this channel is given by

$$C = \frac{1}{2}\int_{-\infty}^{\infty} \log\left(1 + \frac{P(f)|C(f)|^2}{\mathcal{S}_n(f)}\right) df \tag{14.1–10}$$

where $P(f)$, the the input power spectral density, is selected such that

$$P(f) = \left(K - \frac{\mathcal{S}_n(f)}{|C(f)|^2}\right)^+ \tag{14.1–11}$$

where x^+ is defined by

$$x^+ = \max\{0, x\} \tag{14.1–12}$$

and K is selected such that

$$\int_{-\infty}^{\infty} P(f)\,df = P \tag{14.1–13}$$

The water-filling interpretation of this result states that the input power should be allocated to different frequencies in such a way that more power is transmitted at those frequencies of which the channel exhibits a higher signal-to-noise ratio and less power is sent at the frequencies with poor signal-to-noise ratio. A graphical interpretation of the water-filling process is shown in Figure 14.1–1.

The water-filling argument can be also applied to communication over parallel channels. If N parallel discrete-time AWGN channels have noise powers $N_i, 1 \le i \le N$, and an overall power constraint of P, then the total capacity of the parallel channels is given by

$$C = \frac{1}{2}\sum_{i=1}^{N} \log\left(1 + \frac{P_i}{N_i}\right) \tag{14.1–14}$$

where P_i's are selected such that

$$P_i = (K - N_i)^+ \tag{14.1–15}$$

subject to

$$\sum_{i=1}^{N} P_i = P \tag{14.1–16}$$

In addition to frequency selectivity which can be treated through water-filling arguments, a fading channel is characterized with time variations in channel characteristics,

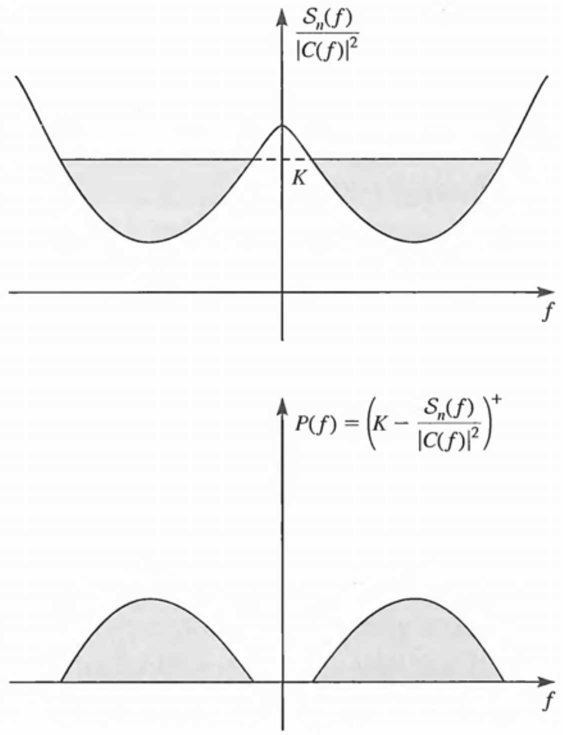


FIGURE 14.1–1
The water-filling interpretation of the channel capacity.

i.e., time selectivity. Since the capacity is defined in the limiting sense as the block length of the code tends to infinity, we can always argue that even in a slowly fading channel the block length can be selected large enough that in any block the channel experiences all possible states, and hence the time averages over one block are equal to the statistical averages. However, from a practical point of view, this would introduce a large delay which is not acceptable in many applications, for instance, speech communication on cellular phones. Therefore, for a delay-constrained system on a slowly fading channel, the ergodicity assumption is not valid.

A common practice to break the inherent memory in fading channels is to employ long interleavers that spread a code sequence across a long period of time, thus making individual symbols experience independent fading. However, employing long interleavers would also introduce unacceptable delay in many applications. These observations make it clear that the notion of capacity is more subtle in the study of fading channels, and depending on the coherence time of the channel and the maximum delay acceptable in the application under study, different channel models and different notions of channel capacity need to be considered. Since fading channels can be modeled as channels whose state changes, we first study the capacity of these channels.

14.1–1 Capacity of Finite-State Channels

A finite-state channel is a channel model for a communication environment that varies with time. We assume that in each transmission interval the state of the channel is selected independently from a set of possible states according to some probability

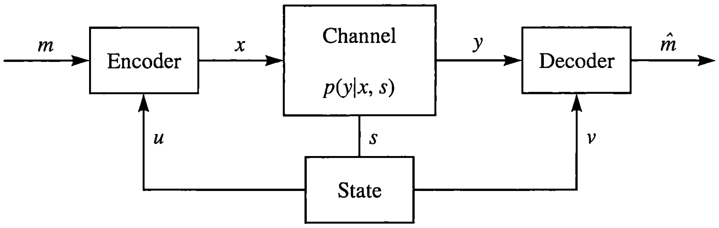


FIGURE 14.1–2
A finite-state channel.

distribution on the space of channel states. The model for a finite-state channel is shown in Figure 14.1–2.

In this channel model, in each transmission the output $y \in \mathscr{Y}$ depends on the input $x \in \mathscr{X}$ and the state of the channel $s \in \mathscr{S}$ through the conditional PDF $p(y|x, s)$. The sets $\mathscr{X}$, $\mathscr{Y}$, and $\mathscr{S}$ denote the input, the output, and the state alphabets, respectively, and are assumed to be discrete sets. The state of the channel is generated independent of the channel input according to

$$p(\boldsymbol{s}) = \prod_{i=1}^{n} p(s_i) \tag{14.1–17}$$

and the channel is memoryless, i.e,

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{s}) = \prod_{i=1}^{n} p(y_i|x_i, s_i) \tag{14.1–18}$$

The encoder and the decoder have access to noisy versions of the state denoted by $u \in \mathscr{U}$ and $v \in \mathscr{V}$, respectively. Based on an original idea of Shannon (1958), Salehi (1992), and Caire and Shamai (1999) have shown that the capacity of this channel can be given as

$$C = \max_{p(t)} I(T; Y|V) \tag{14.1–19}$$

In this expression the maximization is over $p(t)$, the set of all probability mass functions on $\mathscr{T}$ where $\mathscr{T}$ denotes the set of all vectors of length $|\mathscr{U}|$ with components from $\mathscr{X}$. The cardinality of the set $\mathscr{T}$ is $|\mathscr{X}|^{|\mathscr{U}|}$, and the set $\mathscr{T}$ is called the set of *input strategies*.

In the study of fading channels, certain cases of this channel model are of particular interest. The special case where $U = S$ and V is a degenerate random variable corresponds to the case when complete *channel state information* (CSI) is available at the receiver and no channel state information is available at the transmitter. In this case the capacity reduces to

$$C = \max_{p(x)} I(X; Y|S) \tag{14.1–20}$$

where

$$p(s, x, y) = p(s)p(x)p(y|x, s) \tag{14.1–21}$$

Note that since

$$I(X; Y|S) = \sum_{s} p(s)I(X; Y|S = s) \tag{14.1–22}$$

the capacity can be interpreted as the maximum over all input distributions of the average of the mutual information over all channel states. A second interesting case occurs when the state information is available at both the transmitter and the receiver. In this case

$$C = \max_{p(x|s)} I(X; Y|S) = \sum_{s} p(s) \max_{p(x|s)} I(X; Y|S = s) \tag{14.1–23}$$

where the maximization is on all joint probabilities of the form

$$p(s, x, y) = p(s)p(x|s)p(y|x, s) \tag{14.1–24}$$

Clearly since in this case the state information is available at the transmitter, the encoder can choose the input distribution based on the knowledge of the state. Since for each state of the channel the input distribution is selected to maximize the mutual information in that state, the channel capacity is the expected value of the capacities. A third interesting case occurs when complete channel information is available at the receiver but the receiver transmits only a deterministic function of it to the transmitter. In this case $v = s$ and $u = g(s)$, where $g(\cdot)$ denotes a deterministic function. In this case the capacity is given by [see Caire and Shamai (1999)]

$$C = \sum_{u} p(u) \max_{p(x|u)} I(X; Y|S, U = u) \tag{14.1–25}$$

This case corresponds to when the receiver can estimate the channel state but due to communication constraints over the feedback channel can transmit only a quantized version of the state information to the transmitter.

The underlying memoryless assumption in these cases makes these models appropriate for a fully interleaved fading channel.

14.2
ERGODIC AND OUTAGE CAPACITY

To study the difference between ergodic and outage capacity, consider the two-state channel shown in Figure 14.2–1. In this figure two binary symmetric channels, one with crossover probability $p = 0$ and one with crossover probability $p = 1/2$, are shown. We consider two different channel models based on this figure.

1. In channel model 1 the input and output switches choose the top channel (BSC 1) with probability δ and the bottom channel (BSC 2) with probability $1 - \delta$, *independently for each transmission*. In this channel model each symbol is transmitted independently of the previous symbols, and the state of the channel is also selected independently for each symbol.
2. In channel model 2 the top and the bottom channels are selected at the beginning of the transmission with probabilities δ and $1 - \delta$, respectively; but once a channel is selected, it will not change for the entire transmission period.

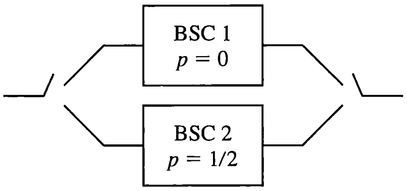


FIGURE 14.2–1
A two-state channel.

From Chapter 6 we know that the capacities of the top and bottom channels are $C_1 = 1$ and $C_2 = 0$ bits per transmission, respectively. To find the capacity of the first channel model, we note that since in this case for transmission of each symbol the channel is selected independently over a long block, the channel will experience both BSC component channels according to their corresponding probabilities. In this case time and ensemble averages can be interchanged, the notion of *ergodic capacity*, denoted by $\overline{C}$, applies, and the results of the preceding section can be used. The capacity of this channel model depends on the availability of the state information. We distinguish three cases for the first channel model.

1. Case 1: No channel state information is available at the transmitter or receiver. In this case it is easy to verify that the average channel is a binary symmetric channel with crossover probability of $\frac{1-\delta}{2}$, and hence the ergodic capacity is

$$\overline{C} = 1 - H_b\left(\frac{1-\delta}{2}\right) \tag{14.2–1}$$

2. Case 2: Channel state information available at the receiver. Using Equation 14.1–22, we observe that in this case we maximize the mutual information with a *fixed* input distribution. But since regardless of the state of the channel a uniform input distribution maximizes the mutual information, the ergodic capacity of the channel is the average of the two capacities, i.e.,

$$\overline{C} = \delta C_1 + (1-\delta)C_2 = \delta \tag{14.2–2}$$

3. Case 3: Channel state information is available at the transmitter and the receiver. Here we use Equation 14.1–23 to find the channel capacity. In this case we can maximize the mutual information individually for each state, and the capacity is the average of the capacities as given in Equation 14.2–2.

A plot of the two capacities as a function of δ is given in Figure 14.2–2. Note that in this particular channel since the capacity achieving input distribution for the two channels states is the same, the results of cases 2 and 3 are the same. In general the capacities in these cases are different, as shown in Problem 14.7.

In the second channel model where one of the two channels BSC 1 or BSC 2 is selected only once and then used for the entire communication situation, the capacity in the Shannon sense is zero. In fact it is not possible to communicate reliably over this channel model at any positive rate. The reason is that if we transmit at a rate $R > 0$ and channel BSC 2 is selected, the error probability cannot be set arbitrarily small. Since channel BSC 2 is selected with a probability of $1 - \delta > 0$, reliable communication at any rate $R > 0$ is impossible. In fact in this case the channel capacity is a binary random variable which takes values of 1 and 0 with probabilities δ and $1 - \delta$, respectively. This

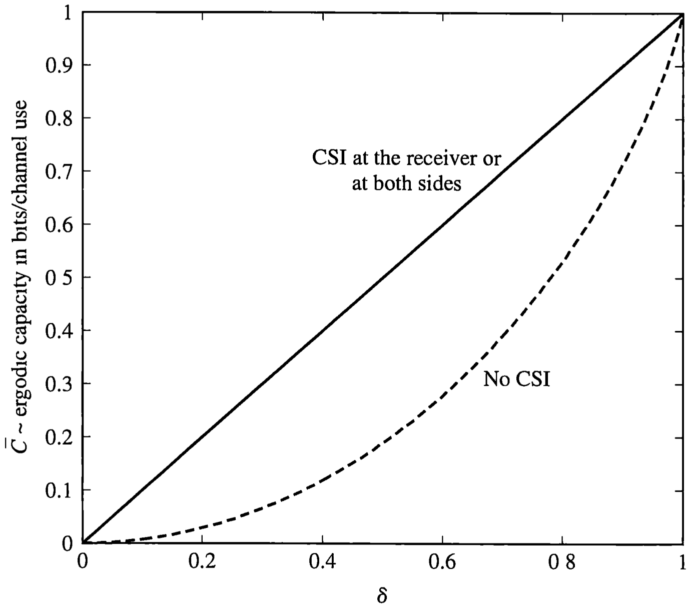


FIGURE 14.2–2
The ergodic capacity of channel model 1.

is a case for which ergodic capacity is not applicable and a new notion of capacity called *outage capacity* is more appropriate (Ozarow et al. (1994)).

We note that since the channel capacity in this case is a random variable, if we transmit at a rate $R > 0$, there is a certain probability that the rate exceeds the capacity and the channel will be in *outage*. The probability of this event is called the *outage probability* and is given by

$$\mathrm{P}_{\mathrm{out}}(R) = \mathrm{P}\left[C < R\right] = F_C(R^-) \tag{14.2–3}$$

where $F_C(c)$ denotes the CDF of the random variable C and $F_C(R^-)$ is the limit-from-left of $F_C(c)$ at point $c = R$.

For any $0 \leq \epsilon < 1$ we can define C_ϵ, the ϵ*-outage capacity* of the channel, as the highest transmission rate that keeps the outage probability under ϵ, i.e.,

$$C_\epsilon = \max\left\{R : \mathrm{P}_{\mathrm{out}}(R) \leq \epsilon\right\} \tag{14.2–4}$$

In the channel model 2, the ϵ-outage capacity of the channel is given by

$$C_\epsilon = \begin{cases} 0 & \text{for } 0 \leq \epsilon < 1 - \delta \\ 1 & \text{for } 1 - \delta \leq \epsilon < 1 \end{cases} \tag{14.2–5}$$

14.2–1 The Ergodic Capacity of the Rayleigh Fading Channel

In this section we study the ergodic capacity of the Rayleigh fading channel. The underlying assumption is that the channel coherence time and the delay restrictions of the channel are such that perfect interleaving is possible and the discrete-time equivalent of the channel can be modeled as a memoryless AWGN channel with independent Rayleigh channel coefficients. The lowpass discrete-time equivalent of this channel is described by an input-output relation of the form

$$y_i = R_i x_i + n_i \tag{14.2–6}$$

where x_i and y_i are the complex input and output of the channel, R_i is a complex iid random variable with Rayleigh distributed magnitude and uniform phase, and n_i's are iid random variables drawn according to $\mathcal{CN}(0, N_0)$. The PDF of the magnitude of R_i is given by

$$p(r) = \begin{cases} \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}} & r > 0 \\ 0 & r \le 0 \end{cases} \tag{14.2–7}$$

We know from Chapter 2, Equations 2.3–45 and 2.3–27, that R^2 is an exponential random variable with expected value $E[R^2] = 2\sigma^2$. Therefore, if $\rho = |R_i|^2$, then from Equation 2.3–27 we have

$$p(\rho) = \begin{cases} \frac{1}{2\sigma^2} e^{-\frac{\rho}{2\sigma^2}} & \rho > 0 \\ 0 & \rho \le 0 \end{cases} \tag{14.2–8}$$

and since the received power is proportional to ρ, we have

$$P_r = 2\sigma^2 P_t \tag{14.2–9}$$

where P_t and P_r denote the transmitted and the received power, respectively. In the following discussion we assume that $2\sigma^2 = 1$, thus $P_t = P_r = P$. The extension of the results to the general case is straightforward.

Depending on the availability of channel state information at the transmitter and receiver, we study the ergodic channel capacity in three cases.

No Channel State Information In this case the receiver knows neither the magnitude nor the phase of the fading coefficients R_i; hence no information can be transmitted on the phase of the input signal. The input-output relation for the channel is given by

$$y = Rx + n \tag{14.2–10}$$

where R and n are independent circular complex Gaussian random variables drawn according to $\mathcal{CN}(0, 2\sigma^2)$ and $\mathcal{CN}(0, N_0)$, respectively.

To determine the capacity of the channel in this case, we need to derive an expression for $p(y|x)$ which can be written as

$$p(y|x) = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\infty} p(y|x, r, \theta) p(r)\, dr\, d\theta \tag{14.2–11}$$

where $p(r)$ is given by Equation 14.2–7 and

$$p(y|x, r, \theta) = \frac{1}{\pi N_0} e^{\frac{|y - re^{j\theta}x|^2}{N_0}} \tag{14.2–12}$$

It can be shown (see Problem 14.8) that Equation 14.2–11 simplifies to

$$p(y|x) = \frac{1}{\pi\left(N_0 + |x|^2\right)} e^{-\frac{|y|^2}{N_0+|x|^2}} \tag{14.2–13}$$

This relation clearly shows that all the phase information is lost.

It has been shown by Abou-Faycal et al. (2001) that when an input power constraint is imposed, the capacity achieving input distribution for this case has a *discrete* iid amplitude and an irrelevant phase. However, there exists no closed-form expression for the capacity in this case. Moreover, in the same work it has been shown that for relatively low average signal-to-noise ratios, when P/N_0 is less than 8 dB, only two signal levels, one of them at zero, are sufficient to achieve capacity; i.e., in this case on-off signaling is optimal. As the signal-to-noise ratio decreases, the amplitude of the nonzero input in the optimal on-off signaling increases, and in the limit for $P/N_0 \to 0$ we obtain

$$\overline{C} = \frac{1}{\ln 2}\frac{P}{N_0} \approx 1.44\frac{P}{N_0} \tag{14.2–14}$$

By comparing this result with Equation 14.1–8 it is seen that for low signal-to-noise ratios the capacity is equal to the capacity of an AWGN channel; but at high signal-to-noise ratios the capacity is much lower than the capacity of an AWGN channel.

Although no closed form for the capacity exists, a parametric expression for the capacity is derived in Taricco and Elia (1997). The parametric form of the capacity is given by

$$\begin{aligned} P &= \mu e^{-\gamma-\Psi(\mu)} - 1 \\ \overline{C} &= \frac{\mu - \gamma - \mu\Psi(\mu) - 1}{\ln 2} + \log_2 \Gamma(\mu) \end{aligned} \tag{14.2–15}$$

where $\Psi(z)$ is the *digamma function* defined by

$$\Psi(z) = \frac{\Gamma'(z)}{\Gamma(z)} \tag{14.2–16}$$

and $\gamma = -\Psi(1) \approx 0.5772156$ is *Euler's constant.*

A plot of capacity in this case is shown in Figure 14.2–3. The capacity of AWGN is also given for reference. It is clearly seen that lack of information about the channel state is particularly harmful at high signal-to-noise ratios.

State Information at the Receiver Since in this case the phase of the fading process is available at the receiver, the receiver can compensate for this phase; hence without loss of generality we can assume that fading is modeled by a multiplicative real coefficient R with Rayleigh distribution whose effect on the power is a multiplicative coefficient ρ with exponential PDF. Using Equation 14.1–22, we have to find the expected value of the mutual information over all possible states. This corresponds to finding the expected value of

$$C = \log\left(1 + \rho\frac{P}{N_0}\right) \tag{14.2–17}$$

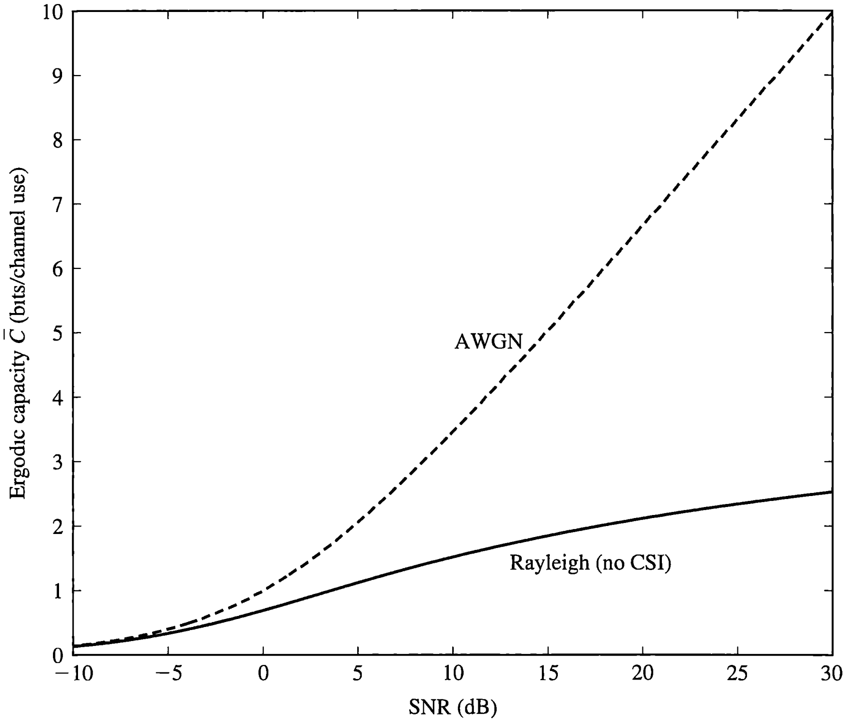


FIGURE 14.2–3
The ergodic capacity of a Rayleigh fading channel with no CSI.

in which ρ has an exponential PDF given by Equation 14.2–8. Since log is a concave function, we can use Jensen's inequality (see Problem 6.29) to show that

$$\begin{aligned}\overline{C} &= \mathrm{E}\left[\log\left(1+\rho\frac{P}{N_0}\right)\right]\\ &\le \log\left(1+\mathrm{E}[\rho]\frac{P}{N_0}\right)\\ &= \log\left(1+\frac{P}{N_0}\right)\end{aligned} \tag{14.2–18}$$

This shows that in this case the capacity is upper-bounded by the capacity of an AWGN channel whose signal-to noise-ratio is equal to the average signal-to-noise ratio of the Rayleigh fading channel.

To find an expression for the capacity in this case, we note that

$$\begin{aligned}\overline{C} &= \int_0^\infty \log\left(1+\rho\frac{P}{N_0}\right)e^{-\rho}\,d\rho\\ &= \frac{1}{\ln 2}e^{\frac{N_0}{P}}\,\Gamma\left(0,\frac{N_0}{P}\right)\\ &= \frac{1}{\ln 2}e^{\frac{1}{\mathrm{SNR}}}\,\Gamma\left(0,\frac{1}{\mathrm{SNR}}\right)\end{aligned} \tag{14.2–19}$$

where $\Gamma(a, z)$ denotes the *complementary gamma function*, defined by

$$\Gamma(a, z) = \int_z^\infty t^{a-1} e^{-t} \, dt \tag{14.2–20}$$

Note that $\Gamma(a, 0) = \Gamma(a)$.

At low SNR values we can use the approximation

$$\log\left(1 + \rho \frac{P}{N_0}\right) \approx \frac{1}{\ln 2} \frac{P}{N_0} \rho \tag{14.2–21}$$

and therefore at low signal-to-noise ratios the capacity is given by

$$\overline{C} \approx \frac{P}{N_0 \ln 2} \int_0^\infty \rho e^{-\rho} \, d\rho \approx 1.44 \, \text{SNR} \tag{14.2–22}$$

which is equal to the capacity of an AWGN channel at low signal-to-noise ratios. At high signal-to-noise ratios we have

$$\log\left(1 + \rho \frac{P}{N_0}\right) \approx \log\left(\rho \frac{P}{N_0}\right) \tag{14.2–23}$$

and the capacity becomes

$$\begin{aligned} \overline{C} &\approx \frac{1}{\ln 2} \int_0^\infty \log\left(\rho \frac{P}{N_0}\right) e^{-\rho} \, d\rho \\ &= \log \text{SNR} + \frac{1}{\ln 2} \int_0^\infty (\ln \rho) e^{-\rho} \, d\rho \\ &= \log \text{SNR} - 0.8327 \end{aligned} \tag{14.2–24}$$

Note that the capacity of an AWGN channel at high signal-to-noise ratios is approximated by log(SNR); therefore at high signal-to-noise ratios, the ergodic capacity of a Rayleigh fading channel with channel state information at the receiver lags the capacity of the AWGN channel by 0.83 bit per complex dimension.

Plots of the capacities of this channel model and the capacity of an AWGN channel with comparable SNR are given in Figure 14.2–4. Unlike the case where no CSI is available, in this case the asymptotic difference between the two curves at high signal-to-noise ratios is roughly 2.5 dB. This compares very favorably with the performance difference of different signaling schemes over Rayleigh fading and AWGN channels. We recall from Equation 13.3–13 that the error probability of common signaling schemes over Rayleigh fading channels decreases inversely with the signal-to-noise ratio, whereas on Gaussian channels the error probability is an exponentially decreasing function of the signal-to-noise ratio. For instance, to achieve an error probability of 10^{-5} using BPSK, an AWGN channel requires a γ_b of 9.6 dB and a Rayleigh fading channel requires 44 dB. This is a huge performance difference. The much lower performance difference between capacities is highly promising and indicates that coding can provide considerable gain in fading channels. The required length of the codewords on fading channels is largely dependent on the dynamics of the fading process and the coherence time of the channel, whereas in an AWGN channel the AWGN effects are averaged over a codeword. In a fading channel, in addition to noise effects, fading effects have

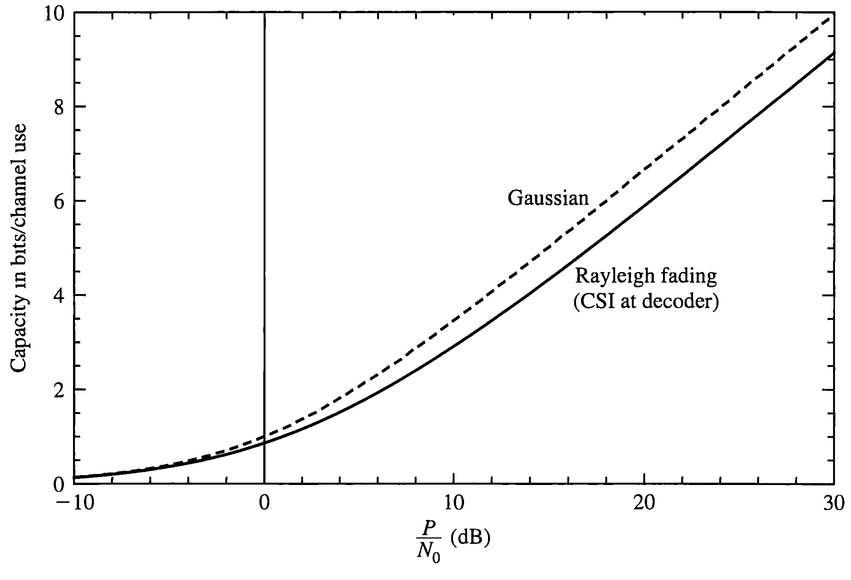


FIGURE 14.2–4
Capacity of Gaussian and Rayleigh fading channel with CSI at the decoder.

to be averaged out over the codeword length. If the channel coherence time is large, this could require very large codeword lengths and could entail unacceptable delay. Interleaving is often used to reduce large codeword requirements, but it cannot reduce the delay in fading channels. Another alternative would be to spread the transmitted code components in the frequency domain to benefit from the diversity. This approach is studied in Section 14.7.

State Information Available at Both Sides If the state information is available at both the transmitter and the receiver, then the result of Equation 14.1–23 can be used. In this case the transmitter can adjust its power level to the fading level similar to the water-filling approach in the frequency domain. Water-filling in time can be employed to allocate the optimal transmitted power as a function of channel state information. Here ρ, the channel state, plays the same role as frequency in the standard water-filling argument, and the capacity is given by

$$\overline{C} = \int_0^\infty \log\left(1 + \rho\,\frac{P(\rho)}{N_0}\right) e^{-\rho}\, d\rho \tag{14.2–25}$$

where $P(\rho)$ denotes the optimum power allocation as a function of the fading parameter ρ. The optimal power allocation is obtained by using water-filling in time, i.e.,

$$\frac{P(\rho)}{N_0} = \left(\frac{1}{\rho_0} - \frac{1}{\rho}\right)^+ \tag{14.2–26}$$

where as before $(x)^+ = \max\{x, 0\}$, and ρ_0 is selected such that

$$\int_0^\infty \left(\frac{1}{\rho_0} - \frac{1}{\rho}\right)^+ e^{-\rho}\, d\rho = \frac{P}{N_0} \tag{14.2–27}$$

Note that from above

$$P(\rho) = \begin{cases} N_0\left(\frac{1}{\rho_0} - \frac{1}{\rho}\right) & \rho > \rho_0 \\ 0 & \rho < \rho_0 \end{cases} \tag{14.2–28}$$

Hence, Equation 14.2–27 becomes

$$\int_{\rho_0}^{\infty} \left(\frac{1}{\rho_0} - \frac{1}{\rho}\right) e^{-\rho}\, d\rho = \frac{P}{N_0} \tag{14.2–29}$$

This equation can be simplified as

$$\frac{e^{-\rho_0}}{\rho_0} - \Gamma(0, \rho_0) = \frac{P}{N_0} \tag{14.2–30}$$

where $\Gamma(a, z)$ is given by Equation 14.2–20. Substituting $P(\rho)$ in the expression for capacity results in

$$\begin{aligned} \overline{C} &= \int_{\rho_0}^{\infty} \log\left(1 + \rho\left(\frac{1}{\rho_0} - \frac{1}{\rho}\right)\right) e^{-\rho}\, d\rho \\ &= \int_{\rho_0}^{\infty} e^{-\rho} \log \frac{\rho}{\rho_0}\, d\rho \\ &= \frac{1}{\ln 2}\Gamma(0, \rho_0) \end{aligned} \tag{14.2–31}$$

Equations 14.2–30 and 14.2–31 provide a parametric description of the capacity of this channel model.

It is interesting to compare the capacity of this channel with an AWGN channel at low and high frequencies. For a very low signal-to-noise ratio, we consider the case where SNR $= 0.1$ corresponding to -10 dB. Substituting this value into Equation 14.2–30 results in $\rho_0 = 1.166$. Substituting this value into Equation 14.2–31 yields $\overline{C} = 0.241$. Computing the capacity of an AWGN channel at SNR $= -10$ dB yields $C = 0.137$. Interestingly, the capacity of the fading channel at low signal-to-noise ratios in this case exceeds the capacity of a comparable AWGN channel. At high signal-to-noise ratios, however, the capacity is less than the capacity of an AWGN channel and is very close to the capacity of a Rayleigh fading channel for which the state information is available only at the receiver.

A plot of capacity of this channel versus the signal-to-noise ratio is given in Figure 14.2–5. The capacity of an AWGN channel is also provided for comparison.

Figure 14.2–6 compares the capacities of Rayleigh fading channels under different availability of state information scenarios with the capacity of the Gaussian channel.

14.2–2 The Outage Capacity of Rayleigh Fading Channels

The outage capacity is considered when due to strict delay restrictions ideal interleaving is impossible and the channel capacity cannot be expressed as the average of the capacities for all possible channel realizations, as was done in the case of the

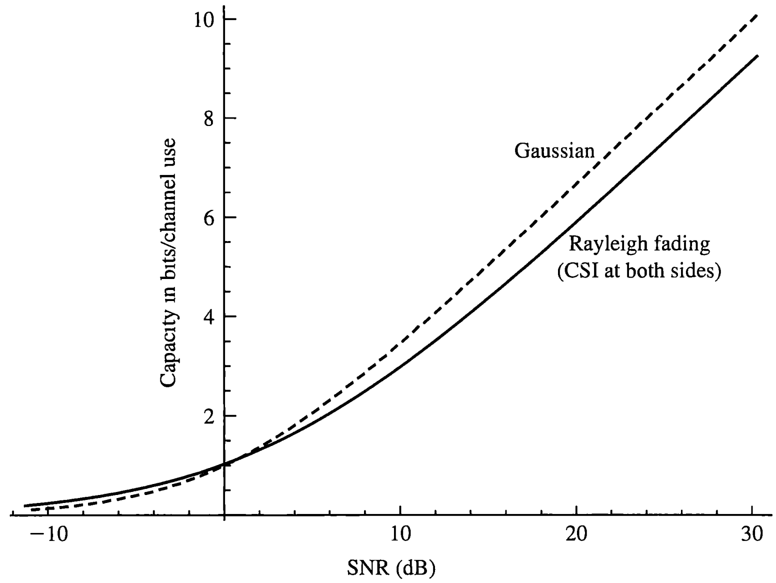


FIGURE 14.2–5
Capacity of Gaussian and Rayleigh fading channel with CSI at both sides.

ergodic capacity. In this case the capacity is a random variable (Ozarow et al. (1994)). We assume at rates less than capacity ideal coding is employed to make transmission effectively error-free. With this assumption, errors occur only when the rate exceeds capacity, i.e., when the channel is in outage.

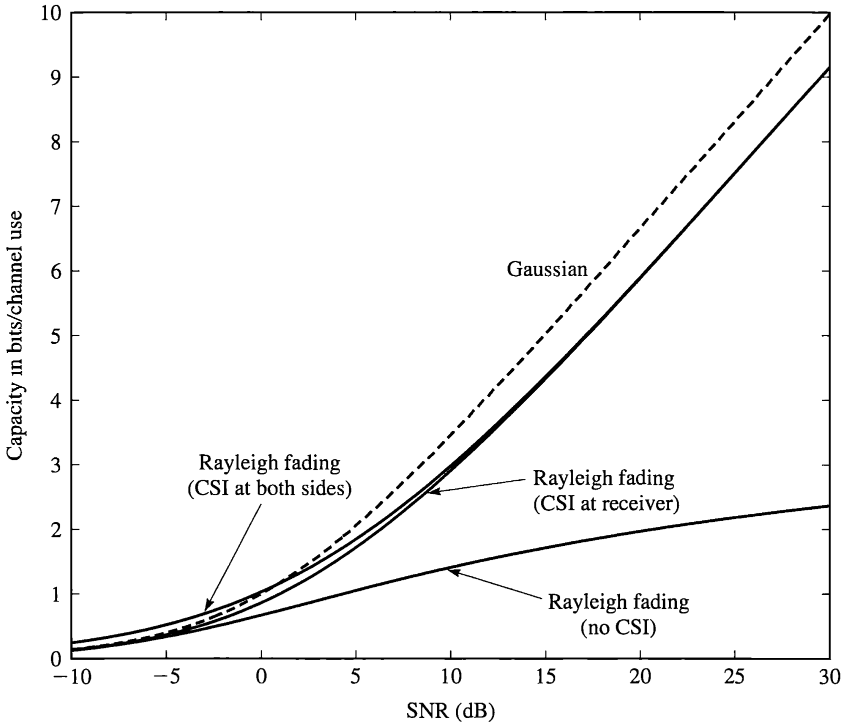


FIGURE 14.2–6
Capacity of Gaussian and Rayleigh fading channel with different CSI.

For a Rayleigh fading channel the outage ϵ-capacity is derived by using Equations 14.2–3 and 14.2–4 as

$$\begin{aligned} C_\epsilon &= \max\{R : \mathrm{P_{out}}(R) \leq \epsilon\} \\ &= \max\{R : F_C(R^-) = \epsilon\} \\ &= F_C^{-1}(\epsilon) \end{aligned} \tag{14.2–32}$$

where $F_C(\cdot)$ is the CDF of the random variable representing the channel capacity.

For a Rayleigh fading channel with normalized channel gain, we have

$$C = \log(1 + \rho\ \mathrm{SNR}) \tag{14.2–33}$$

where ρ is an exponential random variable with expected value equal to 1. The outage probability in this case is given by

$$\mathrm{P_{out}}(R) = \mathrm{P}[C < R] \tag{14.2–34}$$

which simplifies to

$$\begin{aligned} \mathrm{P_{out}}(R) &= \mathrm{P}\left[\rho < \frac{2^R - 1}{\mathrm{SNR}}\right] \\ &= 1 - e^{-\frac{2^R-1}{\mathrm{SNR}}} \end{aligned} \tag{14.2–35}$$

Note that for high signal-to-noise ratios, i.e., for low outage probabilities, this expression can be approximated by

$$\mathrm{P_{out}}(R) \approx \frac{2^R - 1}{\mathrm{SNR}} \tag{14.2–36}$$

Solving for R from Equation 14.2–36 results in

$$R = \log[1 - \mathrm{SNR}\ \ln(1 - \mathrm{P_{out}})] \tag{14.2–37}$$

from which

$$C_\epsilon = \log[1 - \mathrm{SNR}\ \ln(1 - \epsilon)] \tag{14.2–38}$$

We consider the cases of low and high signal-to-noise ratios separately. For low SNR values we have

$$C_\epsilon \approx \frac{\mathrm{SNR}}{\ln 2} \ln \frac{1}{1 - \epsilon} \tag{14.2–39}$$

Since the capacity of an AWGN at low SNR values is $\frac{1}{\ln 2}$ SNR, we conclude that the outage capacity is a fraction of the capacity of an AWGN channel. In fact the capacity of an AWGN channel is scaled by a factor of $\ln \frac{1}{1-\epsilon}$. For instance, for $\epsilon = 0.1$ this value is equal to 0.105, and the outage capacity of the Rayleigh fading channel is only one-tenth of the capacity of an AWGN channel with the same power. For very small ϵ, this factor tends to ϵ and we have

$$C_\epsilon \approx \epsilon C_{\mathrm{AWGN}} \tag{14.2–40}$$

For high signal-to-noise ratios, the capacity is approximated by

$$\begin{aligned} C_\epsilon &\approx \log\left[\text{SNR}\ \ln\frac{1}{1-\epsilon}\right] \\ &= \log \text{SNR} + \log\left(\ln\frac{1}{1-\epsilon}\right) \end{aligned} \tag{14.2–41}$$

The capacity of an AWGN channel at high SNR is log SNR; therefore the outage capacity of the Rayleigh fading channel is less than the capacity of a comparable AWGN channel by $\log\left(\ln\frac{1}{1-\epsilon}\right)$ bits per complex dimension. For $\epsilon = 0.1$ this is equal to 3.25 bits per complex dimension. For very small ϵ we have $\ln\frac{1}{1-\epsilon} \approx \epsilon$, and the difference between the capacities is $\log_2 \epsilon$.

The outage capacity of a Rayleigh fading channel for $\epsilon = 0.1$ and $\epsilon = 0.01$ and the capacity of the AWGN channel are shown in Figure 14.2–7.

Effect of Diversity on Outage Capacity

If a communication system over a Rayleigh fading channel employs L-order diversity, then the random variable $\rho = |R|^2$ has a χ^2 PDF with $2L$ degrees of freedom. In the special case of $L = 1$ we have a χ^2 random variable with two degrees of freedom which is an exponential random variable studied so far. For L-order diversity we use

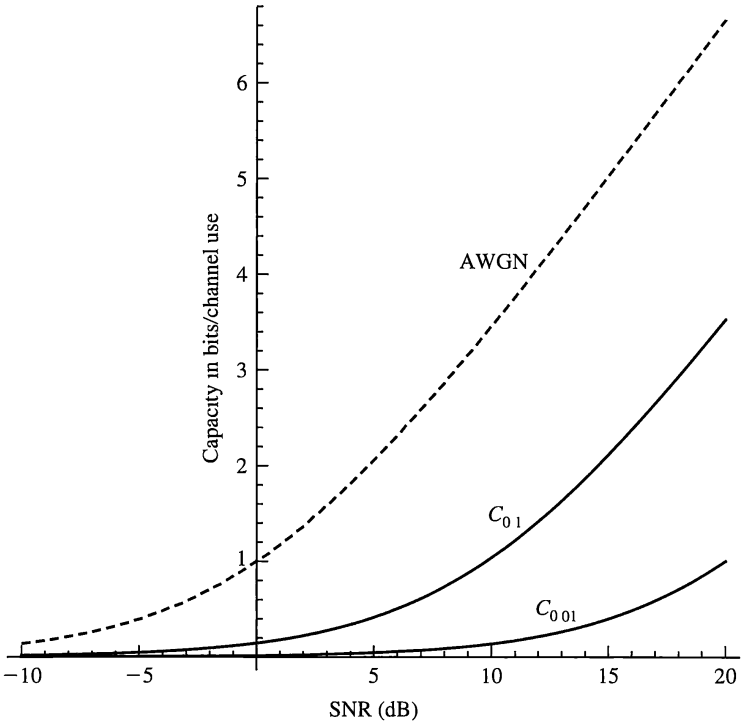


FIGURE 14.2–7
The outage capacity of a Rayleigh fading channel for $\epsilon = 0.1$ and $\epsilon = 0.01$. The capacity of an AWGN channel is given for comparison.

the CDF of a χ^2 random variable given by Equation 2.3–24. We obtain

$$\begin{aligned}
\mathrm{P}_{\text{out}}(R) &= \mathrm{P}\left[\rho < \frac{2^R - 1}{\text{SNR}}\right] \\
&= 1 - e^{-\frac{2^R-1}{\text{SNR}}} \sum_{k=0}^{L-1} \frac{1}{k!}\left(\frac{2^R - 1}{\text{SNR}}\right)^k
\end{aligned} \tag{14.2–42}$$

Equating $\mathrm{P}_{\text{out}}(R)$ to ϵ and solving for R give the ϵ-outage capacity C_ϵ for a channel with L-order diversity. The resulting C_ϵ is obtained by solving the equation

$$e^{-\frac{2^{C_\epsilon}-1}{\text{SNR}}} \sum_{k=0}^{L-1} \frac{1}{k!}\left(\frac{2^{C_\epsilon} - 1}{\text{SNR}}\right)^k = 1 - \epsilon \tag{14.2–43}$$

or equivalently

$$e^{-\frac{2^{C_\epsilon}-1}{\text{SNR}}} \sum_{k=L}^{\infty} \frac{1}{k!}\left(\frac{2^{C_\epsilon} - 1}{\text{SNR}}\right)^k = \epsilon \tag{14.2–44}$$

No closed-form solution for C_ϵ exists for arbitrary L. Plots of $C_{0\ 01}$ for different diversity orders as well as the capacity of an AWGN channel are given in Figure 14.2–8. The noticeable improvement due to diversity is clear from this figure.

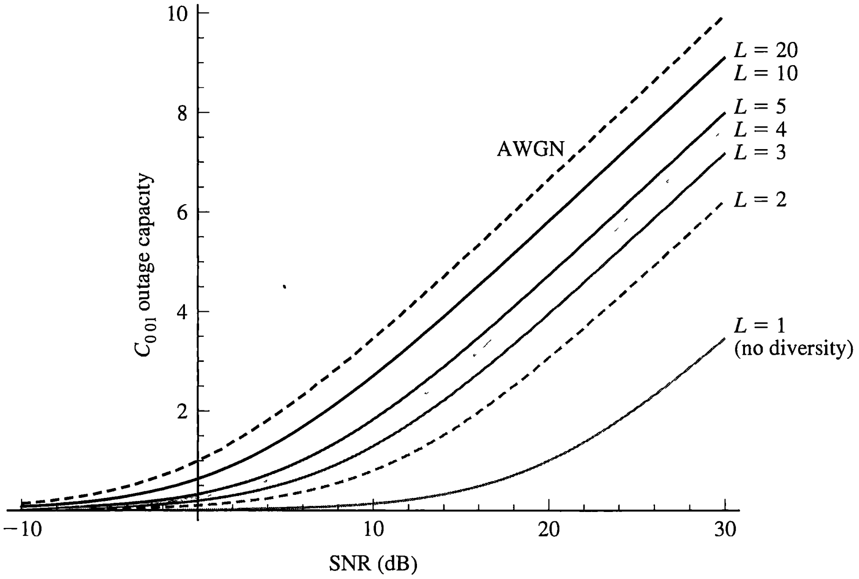


FIGURE 14.2–8
The outage capacity of fading channels with different diversity orders.

14.3

CODING FOR FADING CHANNELS

In Chapter 13 we have demonstrated that diversity techniques are very effective in overcoming the detrimental effects of fading caused by the time-variant dispersive characteristics of the channel. Time and/or frequency diversity techniques may be viewed as a form of repetition (block) coding of the information sequence. From this point of view, the combining techniques described in Chapter 13 represent soft decision decoding of the repetition code. Since a repetition code is a trivial form of coding, we now consider the additional benefits derived from more efficient types of codes. In particular, we demonstrate that coding provides an efficient means of obtaining diversity on a fading channel. The amount of diversity provided by a code is directly related to its minimum distance.

As explained in Section 13.4, time diversity is obtained by transmitting the signal components carrying the same information in multiple time intervals mutually separated by an amount equal to or exceeding the coherence time $(\Delta t)_c$ of the channel. Similarly, frequency diversity is obtained by transmitting the signal components carrying the same information in multiple frequency slots mutually separated by an amount at least equal to the coherence bandwidth $(\Delta f)_c$ of the channel. Thus, the signal components carrying the same information undergo statistically independent fading.

To extend these notions to a coded information sequence, we simply require that the signal waveform corresponding to a particular code bit or code symbol fade independently of the signal waveform corresponding to any other code bit or code symbol. This requirement may result in inefficient utilization of the available time-frequency space, with the existence of large unused portions in this two-dimensional signaling space. To reduce the inefficiency, a number of codewords may be interleaved in time or in frequency or both, in such a manner that the waveforms corresponding to the bits or symbols of a given codeword fade independently. Thus, we assume that the time-frequency signaling space is partitioned into nonoverlapping time-frequency cells. A signal waveform corresponding to a code bit or code symbol is transmitted within such a cell.

In addition to the assumption of statistically independent fading of the signal components of a given codeword, we assume that the additive noise components corrupting the received signals are white Gaussian processes that are statistically independent and identically distributed among the cells in the time-frequency space. Also, we assume that there is sufficient separation between adjacent cells that intercell interference is negligible.

An important issue is the modulation technique that is used to transmit the coded information sequence. If the channel fades slowly enough to allow the establishment of a phase reference, then PSK or DPSK may be employed. In the case where channel state information (CSI) is available at the receiver, knowledge of the phase makes coherent detection possible. If this is not possible, then FSK modulation with noncoherent detection at the receiver is appropriate.

A model of the digital communication system for which the error rate performance will be evaluated is shown in Figure 14.3–1. The encoder may be binary, nonbinary, or a concatenation of a nonbinary encoder with a binary encoder. Furthermore, the code

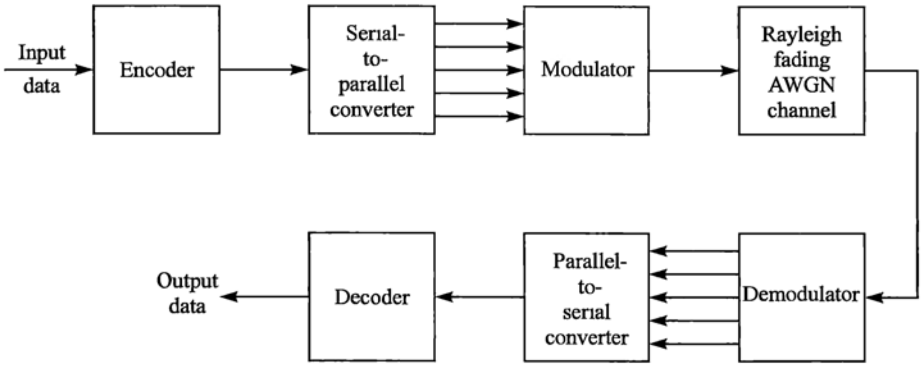

Input data
Encoder
Serial-to-parallel converter
Modulator
Rayleigh fading AWGN channel
Demodulator
Parallel-to-serial converter
Decoder
Output data

FIGURE 14.3–1
Model of communications system with modulation/demodulation and encoding/decoding.

generated by the encoder may be a block code a convolutional code, or, in the case of concatenation, a mixture of a block code and a convolutional code.

To explain the modulation, demodulation, and decoding, consider a linear binary block code in which k information bits are encoded into a block of n bits. For simplicity and without loss of generality, let us assume that all n bits of a codeword are transmitted simultaneously over the channel on multiple frequency/time cells. A codeword $\boldsymbol{c}_i$ having bits $\{c_{ij}\}$ is mapped into signal waveforms and interleaved in time and/or frequency and transmitted. The dimensionality of the signal space depends on the modulation system. For instance, if FSK modulation is employed, each transmitted symbol is a point in the two-dimensional space, hence the dimensionality of the encoded/modulated signal is $2n$. Since each codeword conveys k bits of information, the bandwidth expansion factor for FSK is $B_e = 2n/k$.

The demodulator demodulates the signal components transmitted in independently faded frequency/time cells, providing the sufficient statistics to the decoder which appropriately combines them for each codeword to form the $M = 2^k$ decision variables. The codeword corresponding to the maximum of the decision variables is selected. If hard decision decoding is employed, the optimum maximum-likelihood decoder selects the codeword having the smallest Hamming distance relative to the received codeword.

Although the discussion above assumed the use of a block code, a convolutional encoder can be easily accommodated in the block diagram shown in Figure 14.3–1. For this case the maximum-likelihood soft decision decoding criterion for the convolutional code can be efficiently implemented by means of the Viterbi algorithm. On the other hand, if hard decision decoding is employed, the Viterbi algorithm is implemented with Hamming distance as the metric.

14.4
PERFORMANCE OF CODED SYSTEMS IN FADING CHANNELS

In studying the capacity of fading channels in Section 14.2 we noted that the notion of capacity in fading channels is more involved that the notion of capacity for a standard memoryless channel. The capacity of a fading channel depends on the dynamics of the fading process and how the coherence time of the channels compares with the code length as well as the availability of channel state information at the transmitter and the receiver. In this section we study the performance of a coded system on a fading channel, and we observe that the same factors affect the code performance.

We assume that a coding scheme followed by modulation, or a coded modulation scheme, is employed for data transmission over the fading channel. Our treatment at this point is quite general and includes block and convolutional codes as well as concatenated coding schemes followed by a general signaling (modulation) scheme. This treatment also includes block or trellis-coded modulation schemes.

We assume that M signal space coded sequences $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M\}$ are employed to transmit one of the equiprobable messages $1 \leq m \leq M$. Each codeword $\boldsymbol{x}_i$ is a sequence of n symbols of the form

$$\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in}) \tag{14.4–1}$$

where each x_{ij} is a point in the signal constellation. We assume that the signal constellation is two-dimensional, hence x_{ij}'s are complex numbers.

Depending on the dynamics of fading and availability of channel state information, we can study the effect of fading and derive bounds on the performance of the coding scheme just described.

14.4–1 Coding for Fully Interleaved Channel Model

In this model we assume a very long interleaver is employed and the codeword components are spread over a long interval, much longer than the channel coherence time. As a result, we can assume that the components of the transmitted codeword undergo independent fading. The channel output for this model, when $\boldsymbol{x}_i$ is sent, is given by

$$y_j = R_j x_{ij} + n_j, \qquad 1 \leq j \leq n \tag{14.4–2}$$

where the R_j represents the fading effect of the channel and the n_j is the noise. In this model due to the interleaving, R_j's are independent and n_j's are iid samples drawn according to $\mathcal{CN}(0, N_0)$. The vector input-output relation for this channel is given by

$$\boldsymbol{y} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{n} \tag{14.4–3}$$

where $\boldsymbol{R}$ is an $n \times n$ diagonal matrix

$$\boldsymbol{R} = \operatorname{diag}(R_1, R_2, \ldots, R_n) = \begin{bmatrix} R_1 & 0 & 0 & \cdots & 0 \\ 0 & R_2 & 0 & \cdots & 0 \\ 0 & 0 & R_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \cdots & R_n \end{bmatrix} \tag{14.4–4}$$

and $\boldsymbol{n}$ is a vector with independent n_j's as its components. The R_j's are in general complex, denoting the magnitude and the phase of the fading process.

The maximum-likelihood decoder, having received $\boldsymbol{y}$, uses the rule

$$\hat{m} = \arg\max_{1\le m\le M} p(\boldsymbol{y}|\boldsymbol{x}_m) \tag{14.4–5}$$

to detect the transmitted message m. By the independence of fading and noise components we have

$$p(\boldsymbol{y}|\boldsymbol{x}_m) = \prod_{i=1}^{n} p(y_j|x_{mj}) \tag{14.4–6}$$

The value of $p(y_j|x_{mj})$ depends on the availability of channel state information at the receiver.

CSI Available at the Receiver In this case the output of the channel consists of the output vector $\boldsymbol{y}$ and the channel state sequence $(r_1, r_2, \ldots, r_n)$ which are realizations of random variables $R_1, R_2, \ldots, R_n$, or equivalently the realization of matrix $\boldsymbol{R}$. Therefore, the maximum-likelihood rule, P[observed|input], becomes

$$\prod_{i=1}^{n} p(y_j, r_j|x_{mj}) = \prod_{i=1}^{n} p(r_j)p(y_j|x_{mj}, r_j) \tag{14.4–7}$$

Substituting Equation 14.4–7 into 14.4–5 and dropping the common positive factor $\prod_{i=1}^{n} p(r_j)$ result in

$$\hat{m} = \arg\max_{1\le m\le M} \prod_{i=1}^{n} p(y_j|x_{mj}, r_j) \tag{14.4–8}$$

No CSI Available at the Receiver In this case the ML rule is

$$\hat{m} = \arg\max_{1\le m\le M} \prod_{i=1}^{n} p(y_j|x_{mj}) \tag{14.4–9}$$

where

$$p(y_j|x_{mj}) = \int p(r_i)p(y_j|x_{mj}, r_j)\, dr_i \tag{14.4–10}$$

Performance of Fully Interleaved Fading Channels with CSI at the Receivers A bound on error probability can be obtained by using an approach similar to the one used in Section 6.8–1. Using Equation 6.8–2, we have

$$\begin{aligned} P_{e|m} &\le \sum_{\substack{m'=1\\ m'\neq m}}^{M} \mathrm{P}\left[\boldsymbol{y} \in D_{mm'} | \boldsymbol{x}_m \text{ sent}\right] \\ &= \sum_{\substack{m'=1\\ m'\neq m}}^{M} P_{m\to m'} \end{aligned} \tag{14.4–11}$$

where $P_{m\to m'}$ is the pairwise error probability (PEP), i.e., the probability of error in a binary communication system consisting of two signals $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ when $\boldsymbol{x}_m$ is transmitted. Here we derive an upper bound on the pairwise error probability by using the Chernov bounding technique. For other methods of studying the pairwise error probability, the reader is referred to Biglieri et al. (1995, 1996, 1998a).

A Bound on the Pairwise Error Probability To compute a bound on the PEP, we note that since in this case CSI is available at the receiver, according to Equation 14.4–8, the channel conditional probabilities are $p(y_j|x_{mj}, r_j)$ and hence

$$P_{m\to m'} = \int \mathrm{P}\left[\boldsymbol{x}_m \to \boldsymbol{x}_{m'} \,|\, \boldsymbol{R} = \boldsymbol{r}\right] p(\boldsymbol{r})\, d\boldsymbol{r} \tag{14.4–12}$$

where

$$\begin{aligned} \mathrm{P}\left[\boldsymbol{x}_m \to \boldsymbol{x}_{m'} \,|\, \boldsymbol{R} = \boldsymbol{r}\right] &= \mathrm{P}\left[\ln \frac{p(\boldsymbol{y}|\boldsymbol{x}_{m'}, \boldsymbol{r})}{p(\boldsymbol{y}|\boldsymbol{x}_m, \boldsymbol{r})} > 0\right] \\ &= \mathrm{P}\left[Z_{mm'}(\boldsymbol{r}) > 0\right] \end{aligned} \tag{14.4–13}$$

and the likelihood ratio $Z_{mm'}(\boldsymbol{r})$ becomes

$$\begin{aligned} Z_{mm'}(\boldsymbol{r}) &= \ln \frac{p(\boldsymbol{y}|\boldsymbol{x}_{m'}, \boldsymbol{r})}{p(\boldsymbol{y}|\boldsymbol{x}_m, \boldsymbol{r})} \\ &= \frac{1}{N_0}\left(\|\boldsymbol{y} - \boldsymbol{r}\boldsymbol{x}_m\|^2 - \|\boldsymbol{y} - \boldsymbol{r}\boldsymbol{x}_{m'}\|^2\right) \\ &= \frac{1}{N_0}\sum_{j=1}^{n} Z_{mm'j}(r_j) \end{aligned} \tag{14.4–14}$$

with

$$\begin{aligned} Z_{mm'j}(r_j) &= |y_j - r_j x_{mj}|^2 - |y_j - r_j x_{m'j}|^2 \\ &= |r_j|^2\left(|x_{mj}|^2 - |x_{m'j}|^2\right) + 2\mathrm{Re}\left[y_j^* r_j (x_{m'j} - x_{mj})\right] \end{aligned} \tag{14.4–15}$$

Since we are assuming $\boldsymbol{x}_m$ is transmitted, we have $y_j = r_j x_{mj} + n_j$. Substituting this into Equation 14.4–15 and simplifying yield

$$\begin{aligned} Z_{mm'j}(r_j) &= -|r_j|^2 |x_{mj} - x_{m'j}|^2 - 2\mathrm{Re}\left[r_j n_j^* (x_{mj} - x_{m'j})\right] \\ &= -|r_j|^2 d_{mm'j}^2 - N_j \end{aligned} \tag{14.4–16}$$

where N_j is a real zero-mean Gaussian random variable with variance $2|r_j|^2 d_{mm'j}^2 N_0$ and $d_{mm'j}$ is the Euclidean distance between the constellation points representing the jth components of $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$.

Substituting Equation 14.4–16 into Equation 14.4–13 yields

$$Z_{mm'}(\boldsymbol{r}) = \frac{1}{N_0}\sum_{j=1}^{n}\left(-|r_j|^2 d_{mm'j}^2 - N_j\right) \tag{14.4–17}$$

Using this result, Equation 14.4–13 gives

$$\mathrm{P}\left[\boldsymbol{x}_m \to \boldsymbol{x}_{m'} \,|\boldsymbol{R}=\boldsymbol{r}\right] = \mathrm{P}\left[\sum_{j=1}^{n}\left(|R_j|^2 d^2_{mm'j} + N_j\right) < 0 \,\middle|\, \boldsymbol{R}=\boldsymbol{r}\right] \tag{14.4–18}$$

Applying the Chernov bounding technique discussed in Section 2.4 gives

$$\begin{aligned}\mathrm{P}\left[\sum_{j=1}^{n}\left(|R_j|^2 d^2_{mm'j} + N_j\right) < 0 \,\middle|\, \boldsymbol{R}=\boldsymbol{r}\right] &= \mathrm{E}\left[e^{\nu\sum_{j=1}^{n}\left(|R_j|^2 d^2_{mm'j}+N_j\right)<0} \,\middle|\, \boldsymbol{R}=\boldsymbol{r}\right]\\ &\le \min_{\nu<0}\prod_{j=1}^{n}\mathrm{E}\left[e^{\nu\left(|R_j|^2 d^2_{mm'j}+N_j\right)} \,\middle|\, R_j = r_j\right]\end{aligned} \tag{14.4–19}$$

where $|R_j|$ denotes the envelope of the fading process. Substituting this result into Equation 14.4–12 gives

$$P_{m\to m'} \le \min_{\nu<0}\prod_{j=1}^{n}\int \mathrm{E}\left[e^{\nu\left(|R_j|^2 d^2_{mm'j}+N_j\right)} \,\middle|\, R_j = r_j\right] p(r_j)\,dr_j \tag{14.4–20}$$

Ricean Fading Here we assume that $|R_j|$, the envelope of the fading process, has a Ricean PDF as given by Equation 2.3–56. We can directly apply the result of Example 2.4–2 in Section 2.4, and in particular Equation 2.4–25, to obtain

$$P_{m\to m'} \le \prod_{j=1}^{n}\frac{1}{1+\frac{d^2_{mm'j}}{2N_0}\sigma^2}\exp\left[-\frac{\frac{d^2_{mm'j}}{4N_0}s^2}{1+\frac{d^2_{mm'j}}{2N_0}\sigma^2}\right] \tag{14.4–21}$$

and finally, from Equation 14.4–11 we have

$$P_e \le \frac{1}{M}\sum_{m=1}^{M}\sum_{\substack{m'=1\\ m'\neq m}}^{M}\prod_{j=1}^{n}\frac{1}{1+\frac{d^2_{mm'j}}{2N_0}\sigma^2}\exp\left[-\frac{\frac{d^2_{mm'j}}{4N_0}s^2}{1+\frac{d^2_{mm'j}}{2N_0}\sigma^2}\right] \tag{14.4–22}$$

In Equations 14.4–21 and 14.4–22, σ^2 and s are the parameters of the Ricean random variable determining the envelope of the fading process. The pairwise error probability can also be expressed in terms of the Rice factor K as (see Equation 2.4–26)

$$P_{m\to m'} \le \prod_{j=1}^{n}\frac{K+1}{K+1+\frac{Ad^2_{mm'j}}{4N_0}}\exp\left[-\frac{\frac{AKd^2_{mm'j}}{4N_0}}{K+1+\frac{Ad^2_{mm'j}}{4N_0}}\right] \tag{14.4–23}$$

where $A = \mathrm{E}\left[|R_j|^2\right] = s^2 + 2\sigma^2$ is the fading gain and $K = \frac{s^2}{2\sigma^2}$ is the Rice factor. From Equations 14.4–21 and 14.4–23 it is seen that if for one particular codeword component j we have $x_{mj} = x_{m'j}$, and hence $d_{mm'j} = 0$, the corresponding term in the product is equal to 1. Therefore, it is sufficient to consider only those terms in the product for which $x_{mj} \neq x_{m'j}$. Let us denote the components j for which $x_{mj} \neq x_{m'j}$ by $\mathcal{J}_{mm'}$, i.e,

$$\mathcal{J}_{mm'} = \{1 \leq j \leq n : x_{mj} \neq x_{m'j}\} \tag{14.4–24}$$

Then

$$P_{m \to m'} \leq \prod_{j \in \mathcal{J}_{mm'}} \frac{1}{1 + \frac{d^2_{mm'j}}{2N_0}\sigma^2} \exp\left[-\frac{\frac{d^2_{mm'j}}{4N_0}s^2}{1 + \frac{d^2_{mm'j}}{2N_0}\sigma^2}\right] \tag{14.4–25}$$

and in terms of the Rice factor,

$$P_{m \to m'} \leq \prod_{j \in \mathcal{J}_{mm'}} \frac{K+1}{K + 1 + \frac{Ad^2_{mm'j}}{4N_0}} \exp\left[-\frac{\frac{AKd^2_{mm'j}}{4N_0}}{K + 1 + \frac{Ad^2_{mm'j}}{4N_0}}\right] \tag{14.4–26}$$

For a normalized fading channel which does not change the transmitted energy, we have $\mathrm{E}[|R|^2] = A = 1$, and the pairwise error probability can be bounded by

$$P_{m \to m'} \leq \prod_{j \in \mathcal{J}_{mm'}} \frac{K+1}{K + 1 + \frac{d^2_{mm'j}}{4N_0}} \exp\left[-\frac{\frac{Kd^2_{mm'j}}{4N_0}}{K + 1 + \frac{d^2_{mm'j}}{4N_0}}\right] \tag{14.4–27}$$

Rayleigh Fading and Gaussian Channels For the special case of a Rayleigh fading channel, i.e., in the extreme case of $s = K = 0$, we have

$$P_{m \to m'} \leq \prod_{j \in \mathcal{J}_{mm'}} \frac{1}{1 + \frac{d^2_{mm'j}}{2N_0}\sigma^2} \tag{14.4–28}$$

and for a normalized Rayleigh fading channel for which $2\sigma^2 = 1$ in which the received power is equal to the transmitted power (see Equation 14.2–9) we obtain

$$P_{m \to m'} \leq \prod_{j \in \mathcal{J}_{mm'}} \frac{1}{1 + \frac{d^2_{mm'j}}{4N_0}} \tag{14.4–29}$$

The other extreme of a Ricean channel occurs when $K \to \infty$. In this case the Ricean channel becomes a Gaussian channel. For this case Equation 14.4–27 reduces to

$$P_{m\to m'} \leq \prod_{j\in\mathcal{J}_{mm'}} e^{-\frac{d^2_{mm'j}}{4N_0}} \tag{14.4–30}$$

or

$$P_{m\to m'} \leq e^{-\frac{d^2_{mm'}}{4N_0}} \tag{14.4–31}$$

This is the standard result for a Gaussian channel used in Equation 4.2–72.

High Signal-to-Noise Ratio Approximation At high signal-to-noise ratios when $\frac{Ad^2_{mm'j}}{4N_0} \gg K+1$, the bound in Equation 14.4–26 can be approximated as

$$P_{m\to m'} \lessapprox \prod_{j\in\mathcal{J}_{mm'}} \frac{(K+1)e^{-K}}{\frac{A^2d^2_{mm'j}}{4N_0}} \tag{14.4–32}$$

We define the *Hamming distance* between $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ as the cardinality of the set $\mathcal{J}_{mm'}$; i.e., the number of components at which $\boldsymbol{x}$ and $\boldsymbol{x}_{m'}$ are different.

$$d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'}) = |\mathcal{J}_{mm'}| = \left|\{1 \leq j \leq n : x_{mj} \neq x_{m'j}\}\right| \tag{14.4–33}$$

The *product distance* of a code is defined as

$$\delta^2(\boldsymbol{x}_m, \boldsymbol{x}_{m'}) = \frac{1}{\left(\overline{\mathcal{E}}_s\right)^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})}} \prod_{j\in\mathcal{J}_{mm'}} d^2_{mm'j} \tag{14.4–34}$$

where $\overline{\mathcal{E}}_s$ is the average energy per codeword, given by

$$\overline{\mathcal{E}}_s = \frac{1}{M}\sum_{m=1}^{M} \|\boldsymbol{x}_m\|^2 \tag{14.4–35}$$

Note that with this definition we have factored the effect of the signal energy and have defined the product distance for a normalized code, which is similar to the original code, but has average energy equal to 1. With this definition Equation 14.4–32 can be written as

$$P_{m\to m'} \lessapprox \frac{\left[(1+K)e^{-K}\right]^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})}}{\left(\frac{\overline{\mathcal{E}}_s}{4N_0}\right)^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \delta^2(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–36}$$

or

$$P_{m\to m'} \lessapprox \left[\frac{(1+K)e^{-K}}{\Gamma_{mm'}\frac{\overline{\mathcal{E}}_s}{4N_0}}\right]^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–37}$$

where

$$\Gamma_{mm'} = \left(\delta^2(\boldsymbol{x}_m, \boldsymbol{x}_{m'})\right)^{\frac{1}{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})}} \tag{14.4–38}$$

is the geometric mean of the Euclidean distances of the unequal components of $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$. Note that the signal-to-noise ratio is multiplied by $\Gamma_{mm'}$, which we call the *coding gain* of sequences $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ due to its similarity to the Gaussian case.

Using Equation 14.4–37, in Equation 14.4–22, we obtain the following approximate bound:

$$P_e \lessapprox \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{m'=1 \\ m' \neq m}}^{M} \left[\frac{(1+K)e^{-K}}{\Gamma_{mm'} \frac{\mathcal{E}_s}{4N_0}} \right]^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–39}$$

For reasonably high signal-to-noise ratios, the dominating term in Equation 14.4–39 is the term corresponding to the codewords with the minimum Hamming distance. In this case we have

$$P_e \lessapprox (M-1) \left[\frac{(1+K)e^{-K}}{\Gamma_{\min} \frac{\mathcal{E}_s}{4N_0}} \right]^{d_{\min}} \tag{14.4–40}$$

where $d_{\min}$ is the minimum Hamming distance of the code and

$$\Gamma_{\min} = \left(\delta^2_{\min}\right)^{\frac{1}{d_{\min}}} \tag{14.4–41}$$

where $\delta^2_{\min}$ denotes the minimum of the product distances of the codeword pairs having the minimum Hamming distance.

For a Rayleigh fading channel $K = 0$ and for high signal-to-noise ratios, Equations 14.4–36, 14.4–37, 14.4–39, and 14.4–40 simplify to

$$P_{m \to m'} \lessapprox \frac{1}{\left(\frac{\mathcal{E}_s}{4N_0}\right)^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \delta^2(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–42}$$

$$P_{m \to m'} \lessapprox \left[\frac{1}{\Gamma_{mm'} \frac{\mathcal{E}_s}{4N_0}} \right]^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–43}$$

$$P_e \lessapprox \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{m'=1 \\ m' \neq m}}^{M} \left[\frac{1}{\Gamma_{mm'} \frac{\mathcal{E}_s}{4N_0}} \right]^{d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})} \tag{14.4–44}$$

$$P_e \lessapprox (M-1) \left[\frac{1}{\Gamma_{\min} \frac{\mathcal{E}_s}{4N_0}} \right]^{d_{\min}} \tag{14.4–45}$$

Note that in Equations 14.4–40 and 14.4–45 we have been rather conservative to use the factor $(M-1)$. This is with the assumption that all codewords are at minimum distance from the transmitted codeword and certainly results in an upper bound on the error probability. A more realistic bound would be obtained if $(M-1)$ were substituted by the (average) number of codewords at distance $d_{\min}$, i.e., the *multiplicity* of the code denoted by $N_{\min}$.

Diversity Through Coding Since the product distance is defined for a unit-energy constellation, its effect is independent of the signal-to-noise ratio. Its effect on the performance of the coded system is to increase the signal-to-noise ratio, or shift the performance plots by $\Gamma_{\min}$, the coding gain. A very important role is played by the minimum Hamming distance of the code. Comparing Equations 14.4–42 to 14.4–45 with the performance of diversity systems derived in Chapter 13, we note that in coded systems the error probability is proportional to $(\text{SNR})^{-d_{\min}}$ and in a system with L-order diversity the performance is propositional to $(\text{SNR})^{-L}$. We conclude that the effect of coding is similar to the effect of an L-order diversity with $L = d_{\min}$. In other words, a code with minimum distance of $d_{\min}$ provides diversity of order $d_{\min}$. This should be clear by noting that a diversity system is equivalent to transmitting a signal L times, and this is similar to using a repetition code of length L for which $d_{\min} = L$. Coding, however, can provide greater flexibility in choice of the diversity order and can provide coding gain as well. In the context of coding for fading channels, the parameter $d_{\min}$ of a code is usually called the *diversity order* or the *effective length* of the code.

From the above discussion it is clear that the factors affecting the performance of a coded system on a Rayleigh fading channel are quite different from the factors affecting the performance on Gaussian channels. On a Gaussian channel the performance of a coded system is mainly determined by the minimum Euclidean distance of the code. In other words, as long as the Euclidean distance between two codewords is large, it does not matter how this distance is distributed among the code components. In a Rayleigh fading channel, two parameters of the code contribute to its performance. The minimum distance of the code determines the diversity order of the coded system and therefore determines the *slope* of the error probability plots of the coded system. This is the most important factor determining the code performance particularly at high signal-to-noise ratios. A second factor that affects the performance is the product distance of the code whose impact on the performance of the coded system is felt through the coding gain $\Gamma_{\min}$. This effect is an additive effect on the performance plots and results in a *horizontal shift* in performance curves. Since $\Gamma_{\min}$ is the geometric mean of the Euclidean distances of the codeword components over nonequal components, and the geometric mean of positive numbers with a constant sum is maximized when the numbers are equal, we conclude that a good performing code over a Rayleigh fading channel must have all the components different to provide the highest diversity and must have the overall Euclidean distance equally distributed among the codeword components to achieve the highest possible coding gain.

Signal Space Diversity To describe the effect of diversity order of a coded system in a Rayleigh fading channel and see the difference in performance between Rayleigh fading and Gaussian channels, consider the two signal sets given in Figure 14.4–1. The signal constellation (a) is a standard QPSK constellation, and (b) is a rotated version of it. If coding affects only the quadrature component of the transmitted signal, the constellation gets contracted in the vertical direction. Under these conditions the constellation points move to the location denoted by the empty circles. If the fading is quite deep, it is possible that the two constellation points with the same real part collapse into the same point, thus causing considerable error probability. It is clear that under these conditions the constellation shown in Figure 14.4–1(b) performs better than

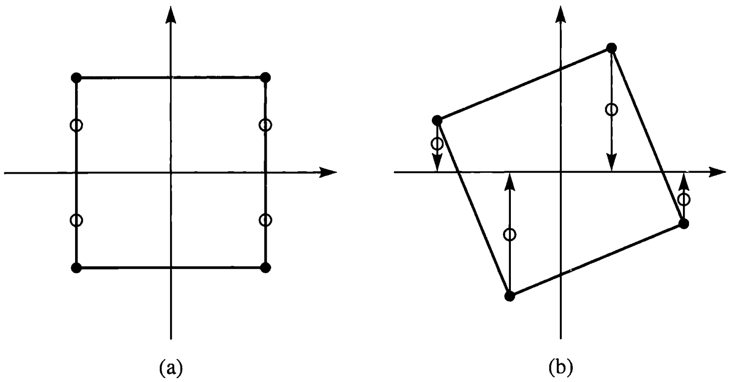


FIGURE 14.4–1
The effect of Hamming distance on the performance of a coded system over fading channels. [*From Boutros and Viterbo (1998), copyright IEEE.*]

the constellation of Figure 14.4–1(a). Note that the two constellations have the same Euclidean distance between signal points, and hence their performance over Gaussian channels is similar. The reason for better performance of constellation (b) is that it has higher Hamming distance and hence provides higher diversity. The diversity order for constellation (a) is 1, whereas the diversity order for constellation (b) is 2. This type of diversity which is a direct result of the choice of the points in the signal space is called *signal space diversity*. Note that in moving from constellation (a) to constellation (b) no redundancy is introduced, and therefore the spectral efficiency of the communication system has not been compromised. The better performance of signal space diversity is achieved by a simple rotation of the constellation. It has been shown by Boutros and Viterbo (1998) that this simple rotation can improve the performance of a QPSK signaling scheme over a Rayleigh fading channel by 8 dB at error probability of 10^{-3}.

Signal space diversity through rotation of a Gaussian constellation can be applied to signal constellations carved from a lattice. Using this technique results in a system with improved performance on fading channels at no bandwidth or power cost. The only drawback of these systems is increased detection complexity when compared with the unrotated lattice. Details on signal space diversity can be found in Boutros et al. (1996) and Boutros and Viterbo (1998).

Performance of Fully Interleaved Fading Channels with No CSI

Derivation of the pairwise error probability in this case is more involved. The details for an MPSK constellation can be found in Divsalar and Simon (1988a) and Jamali and Le-Ngoc (1994). The result for Ricean fading is given by

$$P_{m\to m'} \le \min_{\nu>0} \prod_{j\in\mathcal{J}_{mm'}} e^{\frac{\nu^2}{N_0}|x_{mj}-x_{m'j}|^2} \frac{e^{-K}}{\pi} \int_0^{\pi} \left[1 - 2\sqrt{\pi}\lambda(\theta) Q\left(\sqrt{2}\lambda(\theta)\right) e^{\lambda^2(\theta)}\right] d\theta \tag{14.4–46}$$

where

$$\lambda(\theta) = \frac{\frac{\nu}{2N_0}|x_{mj} - x_{m'j}|^2}{\sqrt{K+1}} - \sqrt{K}\cos(\theta) \tag{14.4–47}$$

At high signal-to-noise ratios and moderate to low values of K, this expression can be further simplified and can be written in the following form

$$P_{m\to m'} \leq \left[(K+1)e^{-K}\frac{\frac{2e}{d_H}}{\Gamma_{mm'}\,\mathrm{SNR}} \cdot \frac{\sum_{j\in\mathcal{J}_{mm'}}|\tilde{\boldsymbol{x}}_m - \tilde{\boldsymbol{x}}_{m'}|^2}{\Gamma_{mm'}}\right]^{d_H} \tag{14.4–48}$$

where $d_H = d_H(\boldsymbol{x}_m, \boldsymbol{x}_{m'})$ is the Hamming distance between $\boldsymbol{x}_m$ and $\boldsymbol{x}_{m'}$ and $\tilde{\boldsymbol{x}}_m = \frac{1}{\sqrt{\mathcal{E}_s}}\boldsymbol{x}_m$ and $\tilde{\boldsymbol{x}}_{m'} = \frac{1}{\sqrt{\mathcal{E}_s}}\boldsymbol{x}_{m'}$. The signal-to-noise ratio is defined as $\mathrm{SNR} = \frac{\mathcal{E}_s}{N_0}$. For the special case of a Rayleigh fading channel for which $K = 0$, this bound becomes

$$P_{m\to m'} \leq \left[\frac{\frac{2e}{d_H}}{\Gamma_{mm'}\,\mathrm{SNR}} \cdot \frac{\sum_{j\in\mathcal{J}_{mm'}}|\tilde{\boldsymbol{x}}_m - \tilde{\boldsymbol{x}}_{m'}|^2}{\Gamma_{mm'}}\right]^{d_H} \tag{14.4–49}$$

14.5
TRELLIS-CODED MODULATION FOR FADING CHANNELS

Our discussion in Section 14.4 shows that in design of good codes for fading channels it is important to consider code parameters that are different from the parameters considered for code design on Gaussian channels. We recall that for code design on Gaussian channels, when soft decision decoding is employed, two parameters determine the performance of the code. These parameters are

1. The minimum Euclidean distance of the code. This is the dominating factor that determines the performance of the code, particularly at high signal-to-noise ratios.
2. The multiplicity of the code, i.e., the number of codewords that are at low Euclidean distance, and particularly at minimum Euclidean distance, from a given codeword. This parameter is particularly important at low signal-to-noise ratios. Turbo codes are examples of codes with low multiplicity that contributes to their excellent performance at low SNRs.

For fading channels the code parameters with highest impact on code performance are

1. The code *diversity order* or *effective length*, given by the minimum Hamming distance of the code. This determines the slope of the error probability plot and is particularly the determining factor at high signal-to-noise ratios.
2. The product distance of the code as defined by Equation 14.4–34 which determines the coding gain defined by Equations 14.4–38 and 14.4–41. This parameter results in a shift in the error probability plot of the code and has the same effect at all signal-to-noise ratios. It is interesting to note that the effect of increasing the product distance on the coding gain is more pronounced at lower diversity orders. This is due to the effect of the $\frac{1}{d_{\min}}$ exponent in Equation 14.4–41. For instance, doubling the product distance in a code with diversity order of 2 increases the coding gain by 1.5 dB, whereas in a code with diversity order of 4, the same increase in the product distance improves the coding gain by 0.75 dB.

3. The multiplicity of the code $N_{\min}$, i.e., the total number of codewords at minimum diversity order and product distance. This factor affects the performance of the code at low signal-to-noise ratios.

14.5–1 TCM Systems for Fading Channels

Trellis-coded modulation was described in Section 8.12 as a means for achieving a coding gain on bandwidth-constrained channels, where we wish to transmit at a bit rate-to-bandwidth ratio $R/W > 1$. For such channels, the digital communication system is designed to use bandwidth-efficient multilevel or multiphase modulation (PAM, PSK, DPSK, or QAM), which allows us to achieve an $R/W > 1$. When coding is applied in signal design for a bandwidth-constrained channel, a coding gain is desired without expanding the signal bandwidth. This goal can be achieved, as described in Section 8.12, by increasing the number of signal points in the constellation over the corresponding uncoded system, to compensate for the redundancy introduced by the code, and designing the trellis code so that the Euclidean distance in a sequence of transmitted symbols corresponding to paths that merge at any node in the trellis is larger than the Euclidean distance per symbol in an uncoded system. In contrast, traditional coding schemes used on fading channels in conjunction with FSK or PSK modulation expand the bandwidth of the modulated signal for the purpose of achieving signal diversity.

In designing trellis-coded signal waveforms for fading channels, we may use the same basic principles that we have learned and applied in the design of conventional coding schemes. In particular, the most important objective in any coded signal design for fading channels is to achieve as large a diversity order as possible.

As indicated above, the candidate modulation methods that achieve high bandwidth efficiency are M-ary PSK, DPSK, QAM, and PAM. The choice depends to a large extent on the channel characteristics. If there are rapid amplitude variations in the received signal, QAM and PAM may be particularly vulnerable, because a wideband automatic gain control (AGC) must be used to compensate for the channel variations. In such a case, PSK or DPSK is more suitable, since the information is conveyed by the signal phase and not by the signal amplitude. DPSK provides the additional benefit that carrier phase coherence is required only over two successive symbols. However, there is an SNR degradation in DPSK relative to PSK.

The discussion and the design criteria provided in Section 14.5 show that a good TCM code for the Gaussian channel is not necessarily a good code for the fading channel. It is quite possible that a trellis code has a large Euclidean distance but has a low effective code length or product distance. In particular some of the good codes designed by Ungerboeck for the Gaussian channel (Ungerboeck (1983)) have parallel branches in their trellises. The existence of parallel branches in TCM codes is due to the existence of uncoded bits, as explained in Chapter 8. Obviously, two paths in the trellis that are similar on all branches but correspond to different branches on a parallel branch have a minimum distance of 1 and provide a diversity order of unity. Such codes are not desirable for transmission over fading channels due to their low diversity order and should be avoided. This is not, however, a problem with the Gaussian channel, and in fact many good TCM schemes that work satisfactorily on Gaussian channels have parallel branches in their trellis representation.

To design TCM schemes with high diversity order, we have to make sure that the paths in the trellis corresponding to different code sequences have long runs of different branches, and the branches are labeled by different symbols from the code constellation. In order for two code sequences to have a diversity order of L, the corresponding paths in the code trellis must remerge at least L branches after diverging, and the two paths on these L branches must have different labels. This clearly indicates that for $L > 1$ parallel transitions have to be excluded.

Let us consider an (n, k, K) convolutional code as shown in Figure 8.1–1. The number of memory elements in this code is Kk, the number of states in the trellis representing this code is $2^{k(K-1)}$, and 2^k branches enter and leave each state of the trellis. Without loss of generality we consider the all-zero path and a path diverging from it. The diverging path from the all-zero path corresponds to an input of k bits that contains at least one 1. Since the number of memory elements of the code is Kk, it takes K sequences of k-bit inputs, all equal to zero, to move the 1 (or 1s) out of the kK memory units, thus bringing back the code to the all-zero state and remerging the path with the all-zero path. This shows that the two paths that have emerged from one state can remerge after at least K branches, and hence this code can potentially provide a diversity order of K. Therefore, the diversity order that a convolutional code can provide is equal to K, the constraint length of the convolutional code. To employ this potential diversity order, we need to have enough points in the signal constellation to assign different signal points to different branches of the trellis.

Let us consider the following trellis code studied by Wilson and Leung (1987). The trellis diagram and the constellation for this TCM scheme are shown in Figure 14.5–1 As seen in the figure, the trellis corresponding to this code is a fully connected trellis, and there are no parallel branches on it, i.e., each branch of the trellis corresponds to a single point in the constellation. The diversity order for this trellis is 2; therefore the error probability is inversely proportional to the square of the signal-to-noise-ratio. The product distance provided by this code is 1.172. It can be easily verified that the squared free Euclidean distance for this code is $d^2_{\text{free}} = 2.586$; therefore the coding

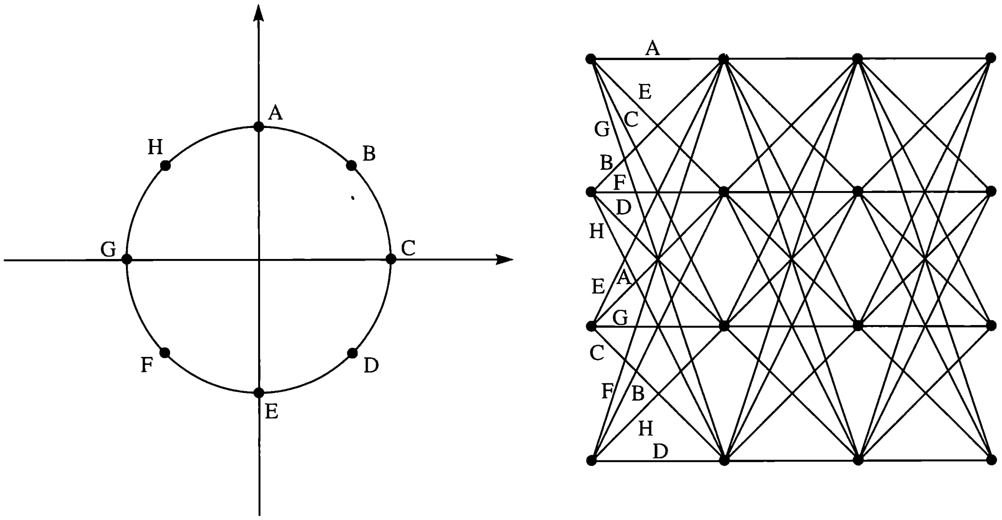


FIGURE 14.5–1
A TCM scheme for fading channels.

gain of the TCM scheme in Figure 14.5–1, when used for transmission over an AWGN channel, is 1.1 dB which is 1.9 dB inferior to the coding gain of the Ungerboeck code of comparable complexity given in Section 8.12.

In Schlegel and Costello (1989) a class of 8-PSK rate 2/3 TCM codes for various constraint lengths is introduced. The search for good codes in this work is done among all codes that can be designed by employing a systematic convolutional code followed by mapping to the 8-PSK signal constellation. It turns out that the advantage of this design procedure is more noticeable at higher constraint lengths. In particular, this design approach results in the same codes obtained by Ungerboeck (1983) when the constraint length is small. At high constraint lengths these codes are capable of providing both higher diversity orders and higher product distances compared to the codes designed by Ungerboeck. For example, for a trellis with 1024 states, these codes can provide a diversity order of 5 and a (normalized) product distance of 128. For comparison, the Ungerboeck code with the same complexity can provide a diversity order of 4 and a product distance of 32.

In Du and Vucetic (1990), Gray coding is employed in the mapping from a convolutional code output to the signal constellation. An exhaustive search is performed on 8-PSK TCM schemes, and it is shown that, particularly at lower constraint lengths, these codes have a better performance compared to those designed in Schlegel and Costello (1989). As the number of states increases, the performance of the codes designed in Schlegel and Costello (1989) is better. As an example for a 32-state trellis code, the approach of Du and Vucetic (1990) results in a diversity order of 3 and a normalized product distance of 32, whereas the corresponding figures for the code designed in Schlegel and Costello (1989) are 3 and 16, respectively.

In Jamali and Le-Ngoc (1991), not only is the design problem of good 4-state 8-PSK trellis codes addressed, but also general design rules are formulated for the Rayleigh fading channel. These design principles can be viewed as the generalization of the design rules formulated in Ungerboeck (1983) for the Gaussian channel. Application of these rules results in improved performance. As an example, by applying these rules one obtains the signal constellation and the trellis shown in Figure 14.5–2.

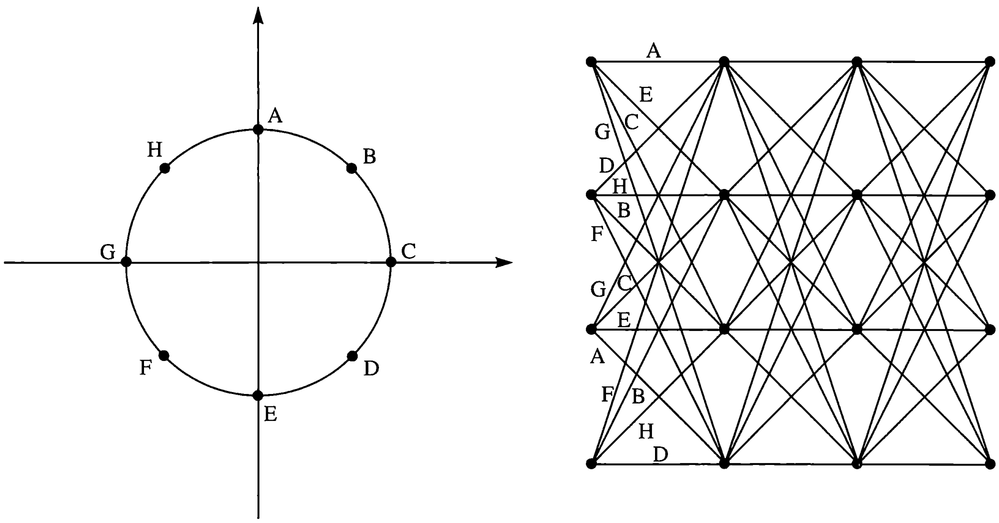


FIGURE 14.5–2
The improved TCM scheme.

It is easy to verify that the coding gain of this code over an AWGN channel (as expressed by the free Euclidean distance) is 2 dB, which is 0.9 dB superior to the code designed in Wilson and Leung (1987) and shown in Figure 14.5–1, and only 1 dB inferior to the Ungerboeck code with a comparable complexity. It is also easy to see that the product distance of this code is twice the product distance of the code shown in Figure 14.5–1, and therefore the performance of this code over a fading channel is superior to the performance of the code designed in Wilson and Leung (1987). Since the squared product distance of this code can be shown to be twice the squared product distance of the code shown in Figure 14.5–1, the asymptotic performance improvement of this code compared to the one designed in Wilson and Leung (1987), when used over fading channels, is $10 \log \sqrt{2} = 1.5$ dB. The encoder for this code can be realized by a convolutional encoder followed by a natural mapping to the 8-PSK signal set.

14.5–2 Multiple Trellis-Coded Modulation (MTCM)

We have seen that the performance of trellis code modulation schemes on fading channels is primarily determined by their diversity order and product distance. In particular, we saw that trellises with parallel branches are to be avoided in transmission over fading channels due to their low (unity) diversity order. In cases where high bit rates are to be transmitted under severe bandwidth restrictions, the signal constellation consists of many signal points. In such cases, to avoid parallel paths in the code trellis, the number of trellis states should be very large, resulting in a very complex decoding scheme.

An innovative approach to avoid parallel branches and at the same time to avoid a very large number of states is to employ *multiple trellis-coded modulation* (MTCM) as first formulated in Divsalar and Simon (1988c). The block diagram for a multiple trellis-coded modulation is shown in Figure 14.5–3.

In the multiple trellis-coded modulation depicted in Figure 14.5–3, at each instance of time $K = km$ information bits enter the trellis encoder and are mapped into $N = nm$ bits, which correspond to m signals from a signal constellation with a total of 2^n signal points, and these m signals are transmitted over the channel. The important fact is that, unlike the standard TCM, here each branch of the trellis is labeled with m signals from the constellation and not only one signal. The existence of more than one

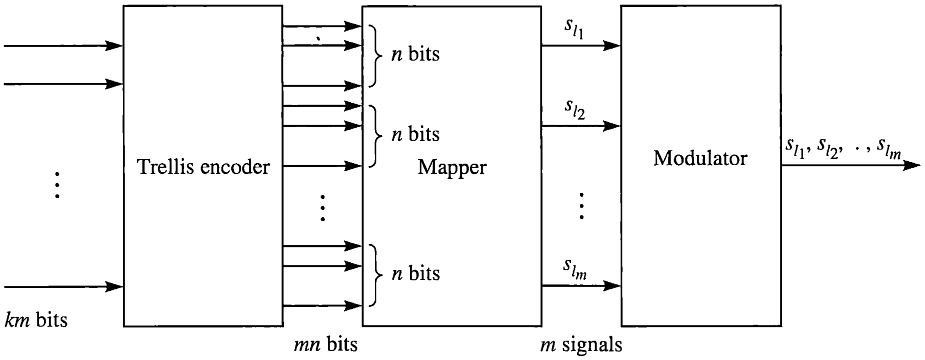


FIGURE 14.5–3
Block diagram of a multiple trellis-coded modulation scheme.

signal corresponding to each trellis branch results in higher diversity order and therefore improved performance when used over fading channels. In fact, MTCM schemes can have a relatively small number of states and at the same time avoid a reduced diversity order. The throughput (or spectral bit rate, defined as the ratio of the bit rate to the bandwidth) for this system is k, which is equivalent to an uncoded (and a conventional TCM) system. In most implementations of MTCM, the value of n is selected to be $k + 1$. Note that with this choice, the case $m = 1$ is equivalent to conventional TCM. The rate of the MTCM code is $R = K/N = k/n$.

In the following example we give a specific TCM scheme and discuss its performance in a fading environment. The signal constellation and the trellis for this example are shown in Figure 14.5–4. For this code we assume $m = 2$, $k = 2$, and $n = 3$. Therefore, the rate of this code is $2/3$, and the trellis selected for the code is a two-state trellis. At each instant of time $K = km = 4$ information bits enter the encoder. This means that there are $2^K = 16$ branches leaving each state of the trellis. Due to the symmetry in the structure of the trellis, there exist eight parallel branches connecting any two states of the trellis. The difference, however, with conventional trellis-coded modulation is that here we assign two signals in the signal space to each branch of the trellis. In fact, corresponding to the $K = 4$ information bits that enter the encoder, $N = nm = 6$ binary symbols leave the encoder. These six binary symbols are used to select two signals from the 8-PSK constellation shown in Figure 14.5–4 (each signal

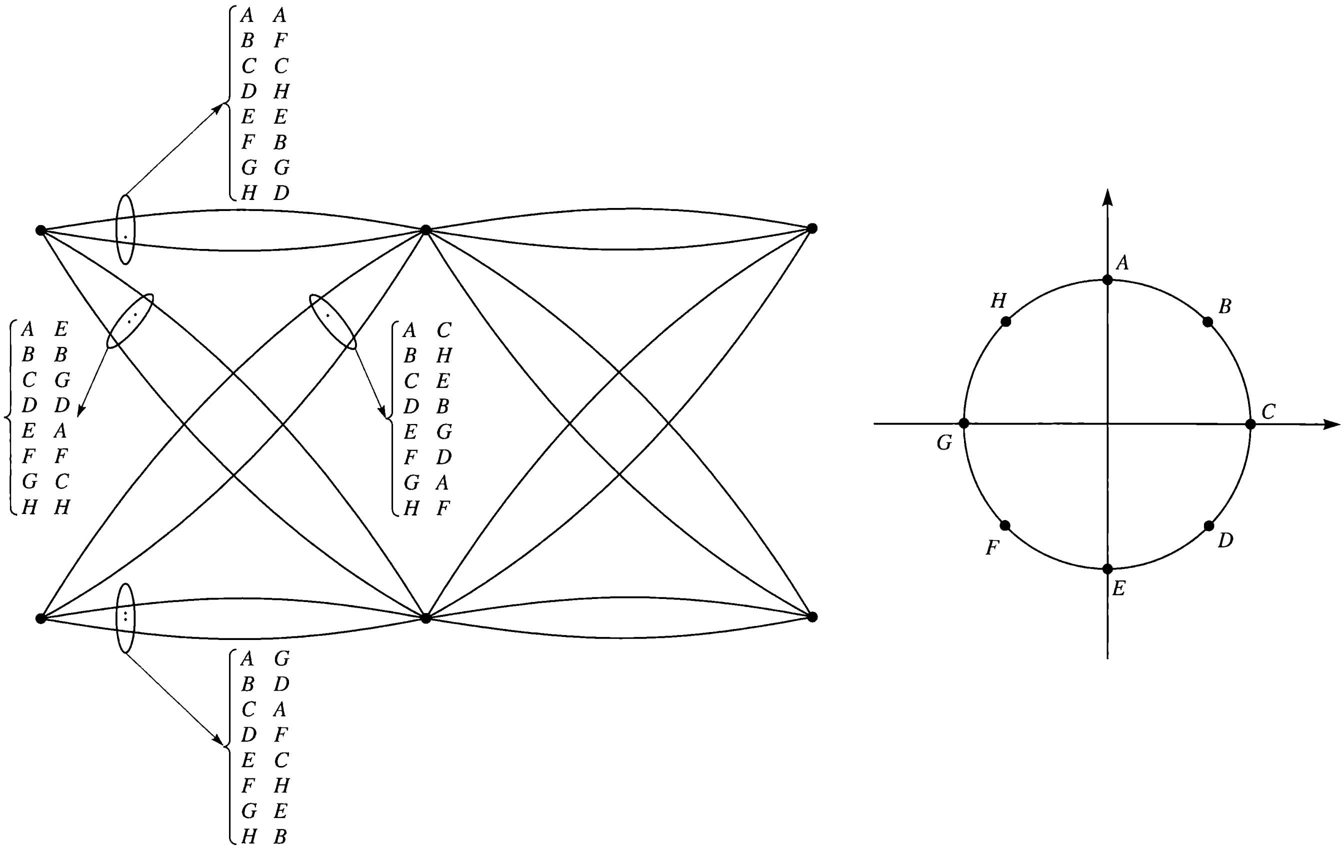


FIGURE 14.5–4
An example of multiple trellis-coded modulation.

requires three binary symbols). The mappings of the branches to the binary symbols are also shown in Figure 14.5–4. Close examination of the mappings suggested in this figure shows that although there exist parallel branches in the trellis for this code, the diversity order provided by this code is equal to 2.

It is seen from the above example that multiple trellis-coded modulation can achieve good diversity, which is essential for transmission through the fading channel, without requiring complex trellises with a large number of states. It can also be shown (see Divsalar and Simon (1988c)), that this same technique can provide all the benefits of using the *asymmetric signal sets,* as described in Divsalar et al. (1987), without the difficulties encountered with time jitter and catastrophic trellis codes. Optimum set partitioning rules for multiple trellis-coded modulation schemes are investigated in Divsalar and Simon (1988b) (see also Biglieri et al. (1991)). It is important to note that the signal set assignments to the trellis branches shown in Figure 14.5–4 are not the best possible signal assignments if this code is to be used over an AWGN channel. In fact, the signal set assignment shown in Figure 14.5–5 provides a performance 1.315 dB superior to the signal set assignment of Figure 14.5–4 when used over an AWGN channel. However, obviously the signal assignment of Figure 14.5–5 can only provide a diversity order equal to unity as opposed to the diversity order of 2 provided by the signal assignment of Figure 14.5–4. This means that on fading channels the performance of the code shown in Figure 14.5–4 is superior to the performance of the code shown in Figure 14.5–5.

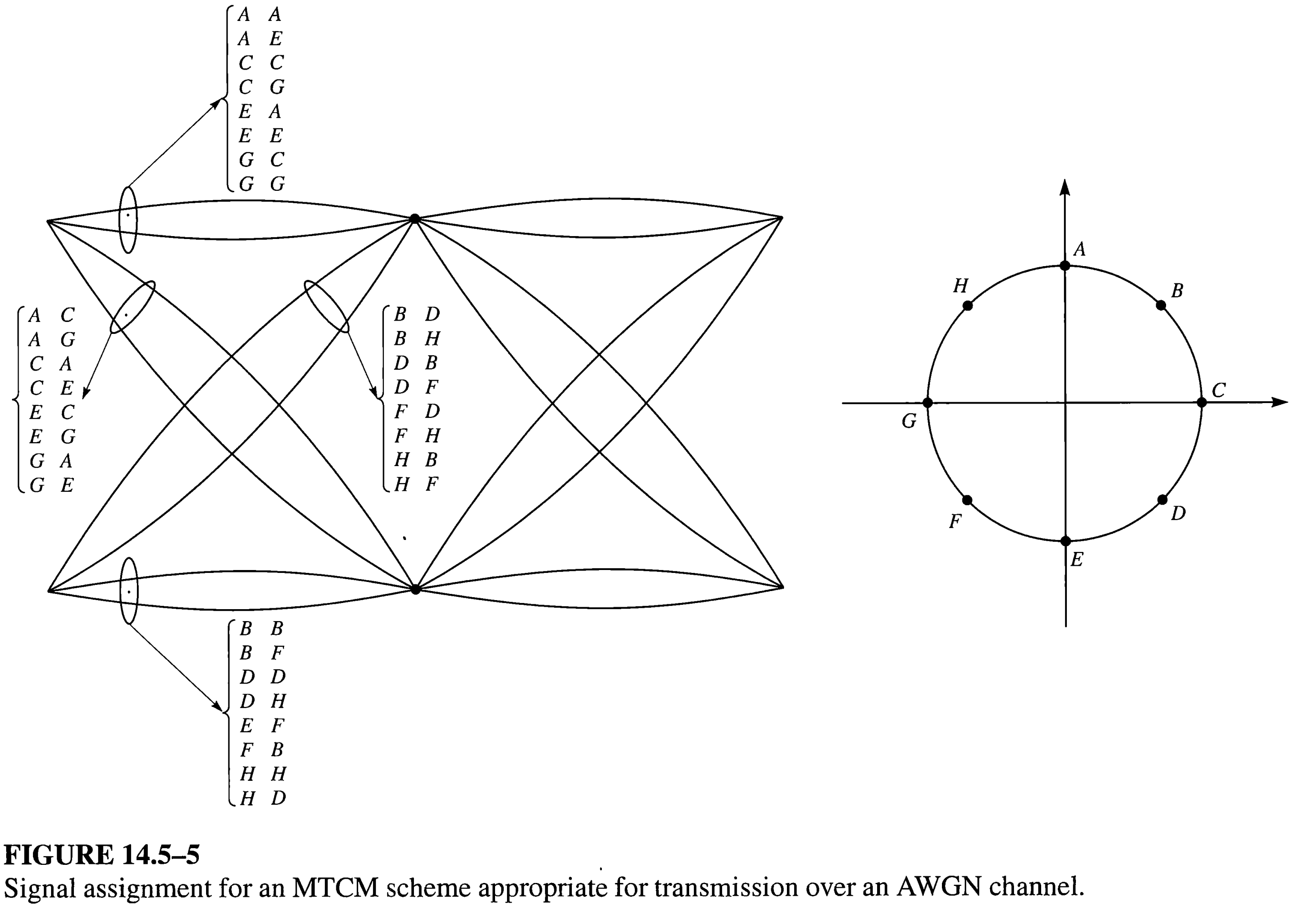


FIGURE 14.5–5
Signal assignment for an MTCM scheme appropriate for transmission over an AWGN channel.

14.6
BIT-INTERLEAVED CODED MODULATION

In Section 8.12 we have seen that a coded modulation system in which coding and modulation are jointly designed as a single entity provides good coding gain over Gaussian channels with no expansion in bandwidth. These codes employ labeling by set partitioning on the code trellis rather than common labeling techniques such as Gray labeling, and these codes achieve their good performance over Gaussian channels by providing large Euclidian distance between trellis paths corresponding to different coded sequences. On the other hand, a code has good performance on a fading channel if it can provide high diversity order, which depends on the minimum Hamming distance of the code, as was seen in Section 14.4–1. For a code to have good performance under both channel models, it has to provide high Euclidean and high Hamming distances. We have previously seen in Chapter 7 that for BPSK and BFSK modulation schemes the relation between Euclidean and Hamming distances is a simple relation given by Equations 7.2–15 and 7.2–17, respectively. These equations indicate that for these modulation schemes Euclidean and Hamming distances are optimized simultaneously.

For coded modulation where expanded signal sets are employed, the relation between Euclidean and Hamming distances is not as simple as the corresponding relations for BPSK and BFSK. In fact, in many coded modulation schemes, where the performance is optimized through labeling the trellis branches by set partitioning using the Ungerboeck's rules (Ungerboeck (1983)), optimal Euclidean distance, and hence optimal performance on the AWGN channels model, is achieved with TCM schemes that have parallel branches and thus have a Hamming distance, and consequently diversity order, equal to unity. These codes obviously cannot perform well on fading channels. In Section 14.5 we gave examples of coded modulation schemes designed for fading channels that achieve good diversity gain on these channels. The underlying assumption in designing these codes was that similar to Ungerboeck's coded modulation approach, the modulation and coding have to be considered as a single entity, and the symbols have to be interleaved by a symbol interleaver of depth usually many times the coherence time of the channel to guarantee maximum diversity. Using symbol interleavers results in the diversity order of the code being equal to the minimum number of distinct symbols between the codewords; and as we have seen in Section 14.5–1, this can be done by eliminating parallel transitions and increasing the constraint length of the code. However, there is no guarantee that the codes using this approach perform well when transmitted over an AWGN channel model. In this section we introduce a coded modulation scheme, called *bit-interleaved coded modulation (BICM)*, that achieves robust performance under both fading and AWGN channel models.

Bit-interleaved coded modulation was first introduced by Zehavi (1992), who introduced a bit interleaver instead of a symbol interleaver at the output of the channel encoder and before the modulator. The idea of introducing a bit interleaver is to make the diversity order of the code equal to the minimum number of distinct bits (rather than channel symbols) by which two trellis paths differ. Using this scheme results in a new soft decision decoding metric for optimal decoding that is different from the metric used in standard coded modulation. A consequence of this approach is that coding and modulation can be done separately. Separate coding and modulation results in a system that is not optimal in terms of achieving the highest minimum Euclidean distance, and therefore the resulting code is not optimal when used on an AWGN channel. However, the diversity order provided by these codes is generally higher than the diversity order of codes obtained by set partitioned labeling and thus provides improved performance over fading channels. A block diagram of a standard TCM system and a bit-interleaved coded modulation system are shown in Figure 14.6–1. In both systems a rate 2/3 convolutional code with an 8-PSK constellation is employed. In the TCM system, the symbol outputs of the encoder are interleaved and then modulated using the 8-PSK constellation and transmitted over the fading channel, in which ρ and n denote the fading and noise processes. In the BICM system, instead of the symbol interleaver we are using three independent bit interleavers that individually interleave the three bit streams. In both systems deinterleavers (at symbol and bit level, respectively) are used at the receiver to undo the effect of interleaving. Note that the fading process (CSI) is available at the receiver in both systems.

Bit-interleaved coded modulation was extensively studied in Caire et al. (1998). This comprehensive study generalized the system introduced by Zehavi (1992), which used multiple bit interleavers at the output of the encoder, and instead used a single bit

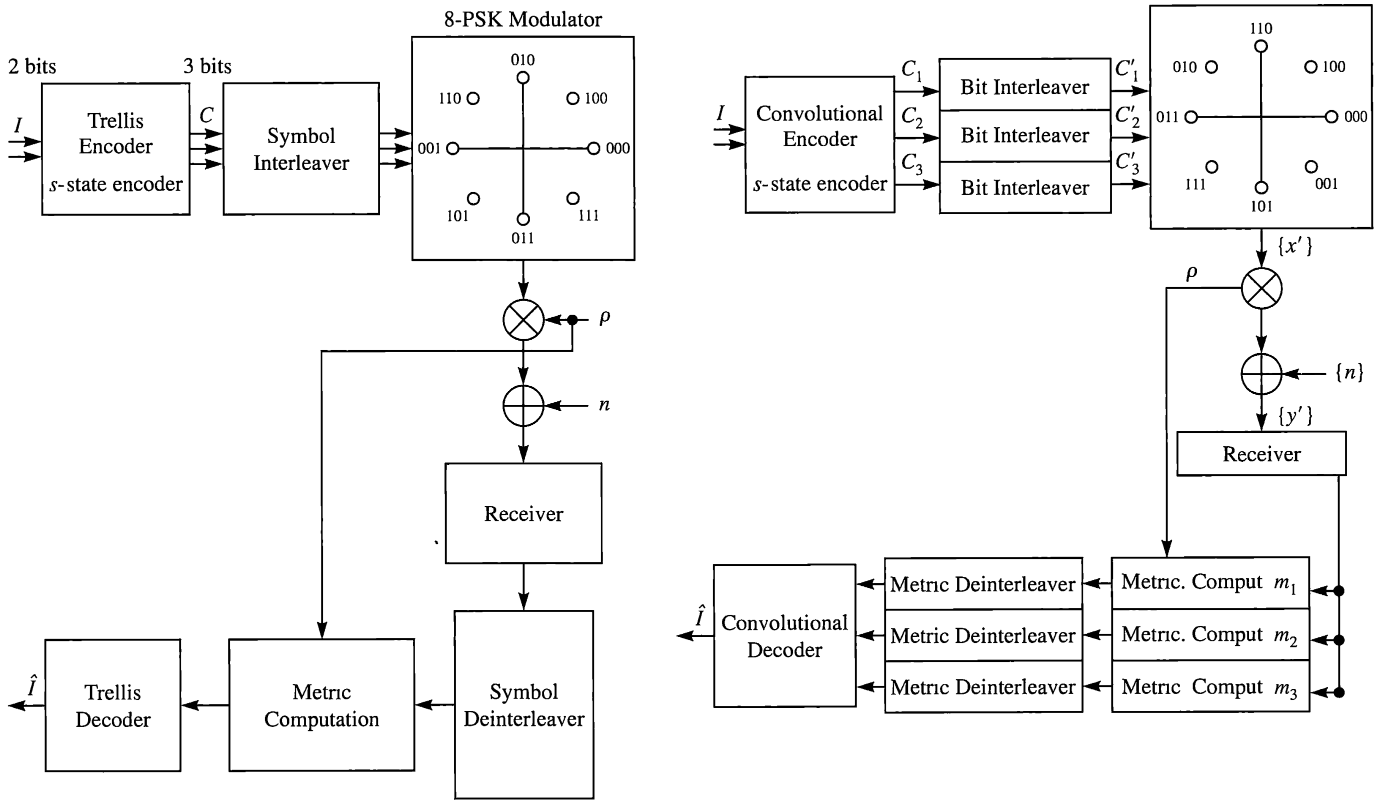

2 bits
3 bits
I
Trellis Encoder
s-state encoder
C
Symbol Interleaver
8-PSK Modulator
010
110
100
001
000
101
011
111
ρ
n
Receiver
Symbol Deinterleaver
Metric Computation
Trellis Decoder
Î
I
Convolutional Encoder
s-state encoder
C1
C2
C3
Bit Interleaver
Bit Interleaver
Bit Interleaver
C'1
C'2
C'3
110
010
100
011
000
111
101
001
{x'}
ρ
{n}
{y'}
Receiver
Metric. Comput m1
Metric. Comput m2
Metric Comput m3
Metric Deinterleaver
Metric Deinterleaver
Metric Deinterleaver
Convolutional Decoder
Î

FIGURE 14.6–1

A TCM system (left) and a BICM system (right). [*From Zehavi (1992) copyright IEEE.*]

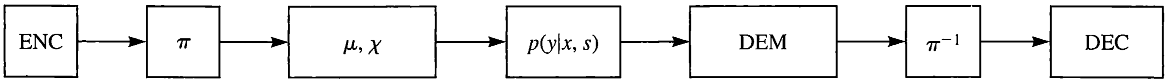


FIGURE 14.6–2
The BICM system studied in Caire et al. (1998). [*From Caire et al. (1998) copyright IEEE.*]

interleaver that operates on the entire encoder output. The block diagram of the system studied in Caire et al. (1998) is shown in Figure 14.6–2.

The encoder output is applied to to an interleaver denoted by π. The output of the interleaver is modulated by the modulator consisting of a label map μ followed by a signal set $\mathcal{X}$. The channel model is a state channel with state s which is assumed to be a stationary, finite-memory vector channel whose input and output symbols $\boldsymbol{x}$ and $\boldsymbol{y}$ are N-tuples of complex numbers. The state $\boldsymbol{s}$ is independent of the channel input $\boldsymbol{x}$, and conditioned on $\boldsymbol{s}$, the channel is memoryless, i.e.,

$$p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{s}) = \prod_{i=1}^{N} p(y_i|\boldsymbol{x}_i, s_i) \tag{14.6–1}$$

The state sequence $\boldsymbol{s}$ is assumed to be a stationary finite-memory random process; i.e., there exists some integer $\nu \geq 0$ such that for all integers r and s and all integers $\nu < k_1 < k_2 < \cdots < k_r$ and $j_1 < j_2 < \cdots < j_s \leq 0$, the sequences $(\boldsymbol{s}_{k_1}, \dots, \boldsymbol{s}_{k_r})$ and $(\boldsymbol{s}_{j_1}, \dots, \boldsymbol{s}_{j_s})$ are independent. The integer ν represents the maximum memory length of the state process. The output of the channel enters the demodulator that computes the branch metrics which after deinterleaving are supplied to the decoder for final decision.

Both coded modulation and BICM systems can be described as special cases of the block diagram of Figure 14.6–2. A coded modulation system results when the encoder is defined over the label alphabet $\mathcal{A}$ and $\mathcal{A}$ and $\mathcal{X} \subset \mathbb{C}^N$ have the same cardinality, i.e., when $|\mathcal{A}| = |\mathcal{X}| = M$. The labeling map $\mu : \mathcal{A} \to \mathcal{X}$ acts on symbol interleaved encoder outputs individually. For Ungerboeck codes the encoder is a rate k/n convolutional code, and $\mathcal{A}$ is the set of binary sequences of length n. The labeling function μ is obtained through applying the set partitioning rules to $\mathcal{X}$.

In BICM, a binary code is employed and its output is bit-interleaved. After interleaving the bit sequence is broken into subsequences of length n, and each is mapped onto a constellation $\mathcal{X} \subset \mathbb{C}^N$ of size $|\mathcal{X}| = M = 2^n$ using a mapping $\mu : \{0, 1\}^n \to \mathcal{X}$.

Let $\boldsymbol{x} \in \mathcal{X}$ and let $\ell^i(\boldsymbol{x})$ denote the ith bit of the label $\boldsymbol{x}$; obviously $\ell^i(\boldsymbol{x}) \in \{0, 1\}$. We define

$$\mathcal{X}_b^i = \{\boldsymbol{x} \in \mathcal{X} : \ell^i(\boldsymbol{x}) = b\} \tag{14.6–2}$$

where $\mathcal{X}_b^i$ denotes the set of all points in the constellation whose label is equal to $b \in \{0, 1\}$ at position i. It can be easily seen that if $\mathrm{P}[b = 0] = \mathrm{P}[b = 1] = 1/2$, then

$$p(\boldsymbol{y}|\ell^i(\boldsymbol{x}) = b, \boldsymbol{s}) = 2^{-(m-1)} \sum_{\boldsymbol{x} \in \mathcal{X}_b^i} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{s}) \tag{14.6–3}$$

The computation of the bit metrics at the demodulator depends on the availability of the channel state information. If CSI is available at the receiver, then the bit metric for the ith bit of the symbol at time k is given by the log-likelihood

$$\lambda^i(\boldsymbol{y}_k, b) = \log \sum_{\boldsymbol{x} \in \mathcal{X}_b^i} p(\boldsymbol{y}_k | \boldsymbol{x}, \boldsymbol{s}) \tag{14.6–4}$$

and for the case with no CSI we have

$$\lambda^i(\boldsymbol{y}_k, b) = \log \sum_{\boldsymbol{x} \in \mathcal{X}_b^i} p(\boldsymbol{y}_k | \boldsymbol{x}) \tag{14.6–5}$$

where $b \in \{0, 1\}$ and $1 \leq i \leq n$. In the bit metric calculation for the no CSI case, we have

$$p(\boldsymbol{y}_k | \boldsymbol{x}) = \int p(\boldsymbol{y}_k | \boldsymbol{x}, \boldsymbol{s}) p(\boldsymbol{s})\, d\boldsymbol{s} \tag{14.6–6}$$

Finally, the decoder uses the ML bit metrics to decode the codeword $\boldsymbol{c} \in \mathcal{C}$ according to

$$\hat{\boldsymbol{c}} = \arg\max_{\boldsymbol{c} \in C} \sum_{i=1}^{N} \lambda^i(\boldsymbol{y}_k, c_k) \tag{14.6–7}$$

which can be implemented using the Viterbi algorithm.

A simpler version of bit metrics can be found using the approximation

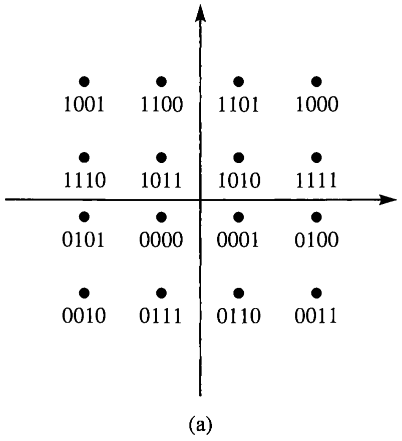
$$\log \sum_i a_i \approx \max_i \log a_i \tag{14.6–8}$$

which is similar to Equation 8.8–33. With this approximation we have the approximate bit metric

$$\tilde{\lambda}^i(\boldsymbol{y}_k, b) = \begin{cases} \max\limits_{\boldsymbol{x} \in \mathcal{X}_b^i} \log p(\boldsymbol{y}_k | \boldsymbol{x}, \boldsymbol{s}) & \text{CSI available} \\ \max\limits_{\boldsymbol{x} \in \mathcal{X}_b^i} \log p(\boldsymbol{y}_k | \boldsymbol{x}) & \text{no CSI} \end{cases} \tag{14.6–9}$$

It turns out that BICM performs better when it is used with Gray labeling as opposed to labeling induced by the set partitioning rules. The Gray and set partitioning labeling for 16-QAM constellation is shown in Figure 14.6–3. Gray labeling is possible for certain constellations. For instance, Gray labeling is not possible for a 32-QAM constellation. In such cases a quasi-Gray labeling achieves good performance.

The channel model for BICM, when ideal interleaving is employed, is a set of n independent memoryless parallel channels with binary inputs that are connected via a random switch to the encoder output. Each channel corresponds to one particular bit position from the total n bits. The capacity and the cutoff rate for this channel model under the assumption of full CSI at the receiver and no CSI are computed in Caire et al. (1998). Figure 14.6–4 shows the cutoff rate for different BICM systems for different QAM signaling schemes over AWGN and Rayleigh fading channels.



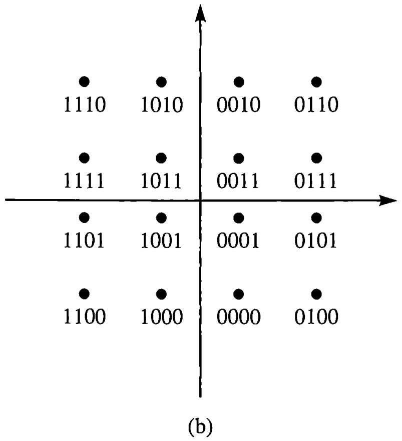
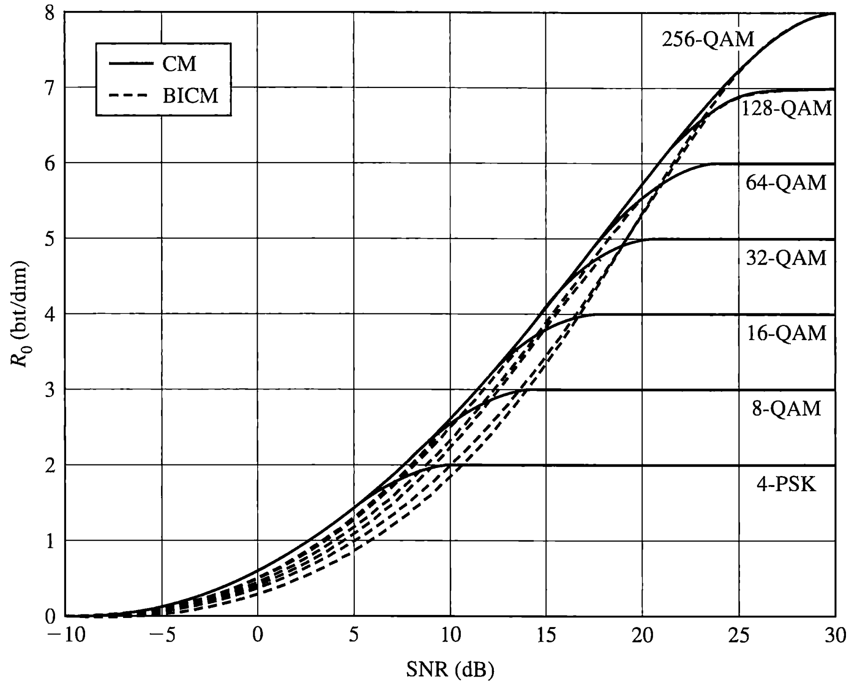


FIGURE 14.6–3
Set partitioning labeling (a) and Gray labeling (b) for 16-QAM signaling. [*From Caire et al. (1998), copyright IEEE.*]

Comparison of these figures shows that for the AWGN channel the performance of coded modulation is superior to the performance of BICM at all signal-to-noise ratios. The performance difference is particularly large for larger constellations and lower-rate codes. For the Rayleigh fading channel BICM outperforms coded modulation at all rates above 1 bit per dimension. The difference in performance is particularly noticeable for larger constellations and higher rates. Similar results can be obtained for orthogonal signals and noncoherent detection.

Table 14.6–1 summarizes the performance parameters of various TCM and BICM schemes with comparable complexity. It is seen that using BICM generally improves the Hamming distance and results in higher diversity order. At the same time BICM marginally reduces the Euclidean distance, resulting in performance deterioration on AWGN channels. This indicates that BICM is a good candidate for channels with variations in the channel model. For instance, Ricean fading channels with varying Rice factor operate somewhere between Rayleigh fading and Gaussian channels. For these channels BICM is an attractive coding scheme displaying robustness to changes in channel characteristics.

For more details on BICM, the interested reader is referred to Caire et al. (1998), Ormeci et al. (2001), Martinez et al. (2006), and Li and Ritcey (1997, 1998, 1999).



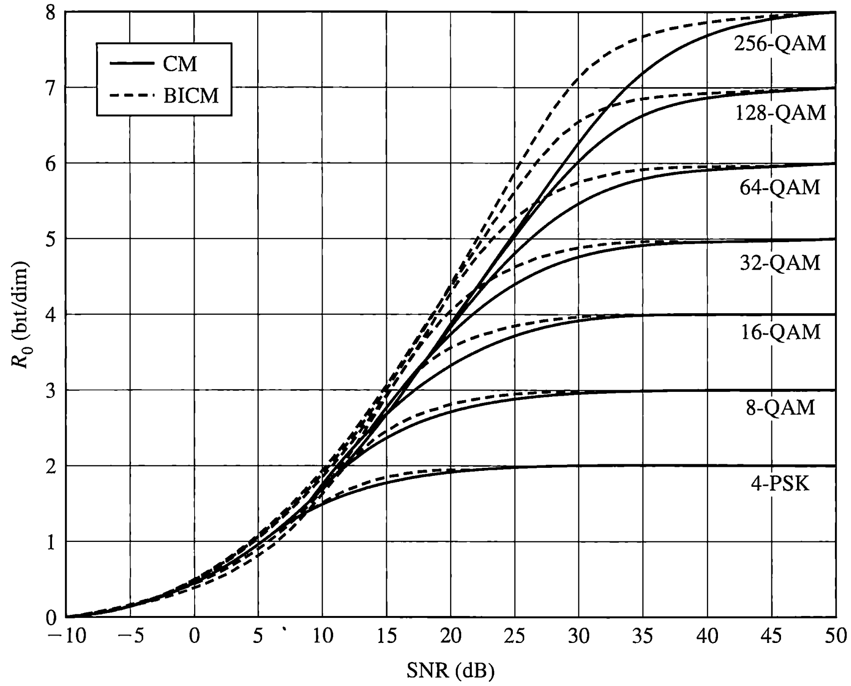


FIGURE 14.6–4

Cutoff rate plots of coded modulation (CM) and BICM for Gray (or quasi-Gray) labeling over AWGN (top) and Rayleigh fading channel (bottom). [*From Caire et al. (1998), copyright IEEE.*]

TABLE 14.6–1
Upper Bounds to Minimum Euclidean Distance and Diversity Order for TCM and BICM for 16-QAM Signaling. Average Energy is Normalized to 1 and Transmission Rate is 3 Bits per Complex Dimension.

Encoder memory	BICM d_E^2	BICM $d_{2(C)}$	TCM d_E^2	TCM $d_{M(C)}$
2	1.2	3	2	1
3	1.6	4	2.4	2
4	1.6	4	2.8	2
5	2.4	6	3.2	2
6	2.4	6	3.6	3
7	3.2	8	3.6	3
8	3.2	8	4	3

Source: From Caire et al. (1998), copyright IEEE

14.7

CODING IN THE FREQUENCY DOMAIN

Instead of bitwise or symbolwise interleaving in the time domain to increase diversity of a coded system and improve the performance over a fading channel, we can achieve similar diversity order by spreading the transmitted signal components in the frequency domain. A candidate modulation scheme for this case is FSK which can be demodulated noncoherently when tracking the channel phase is not possible.

A model for this communication scheme is shown in Figure 14.3–1 where each bit $\{c_{ij}\}$ is mapped into FSK signal waveforms in the following way. If $c_{ij} = 0$, the tone f_{0j} is transmitted; and if $c_{ij} = 1$, the tone f_{1j} is transmitted. This means that $2n$ tones or cells are available to transmit the n bits of the codeword, but only n tones are transmitted in any signaling interval.

The demodulator for the received signal separates the signal into $2n$ spectral components corresponding to the available tone frequencies at the transmitter. Thus, the demodulator can be realized as a bank of $2n$ filters, where each filter is matched to one of the possible transmitted tones. The outputs of the $2n$ filters are detected noncoherently. Since the Rayleigh fading and the additive white Gaussian noises in the $2n$ frequency cells are mutually statistically independent and identically distributed random processes, the optimum maximum-likelihood soft decision decoding criterion requires that these filter responses be square-law-detected and appropriately combined for each codeword to form the $M = 2^k$ decision variables. The codeword corresponding to the maximum of the decision variables is selected. If hard decision decoding is employed, the optimum maximum-likelihood decoder selects the codeword having the smallest Hamming distance relative to the received codeword. Either a block or a convolutional code can be employed as the underlying code in this system.

14.7–1 Probability of Error for Soft Decision Decoding of Linear Binary Block Codes

Consider the decoding of a linear binary (n, k) code transmitted over a Rayleigh fading channel, as described above. The optimum soft-decision decoder, based on the maximum-likelihood criterion, forms the $M = 2^k$ decision variables.

$$\begin{aligned} U_i &= \sum_{j=1}^{n} \left[(1-c_{ij})|\,y_{0j}|^2 + c_{ij}|\,y_{1j}|^2\right] \\ &= \sum_{j=1}^{n} \left[|\,y_{0j}|^2 + c_{ij}\left(|\,y_{1j}|^2 - |\,y_{0j}|^2\right)\right], \qquad i = 1, 2, \ldots, 2^k \end{aligned} \tag{14.7–1}$$

where $|\,y_{rj}|^2$, $j = 1, 2, \ldots, n$, and $r = 0, 1$ represent the squared envelopes at the outputs of the $2n$ filters that are tuned to the $2n$ possible transmitted tones. A decision is made in favor of the code word corresponding to the largest decision variable of the set $\{U_i\}$.

Our objective in this section is the determination of the error rate performance of the soft-decision decoder. Toward this end, let us assume that the all-zero code word $\boldsymbol{c}_1$ is transmitted. The average received signal-to-noise ratio per tone (cell) is denoted by $\bar{\gamma}_c$. The total received SNR for the n tones in $n\bar{\gamma}_c$ and, hence, the average SNR per bit is

$$\bar{\gamma}_b = \frac{n}{k}\bar{\gamma}_c = \frac{\bar{\gamma}_c}{R_c} \tag{14.7–2}$$

where R_c is the code rate.

The decision variable U_1 corresponding to the code word $\boldsymbol{c}_1$ is given by Equation 14.7–1 with $c_{ij} = 0$ for all j. The probability that a decision is made in favor of the mth code word is just

$$\begin{aligned} P_2(m) &= P(U_m > U_1) = P(U_1 - U_m < 0) \\ &= P\left[\sum_{j=1}^{n}(c_{1j} - c_{mj})\left(|\,y_{1j}|^2 - |\,y_{0j}|^2\right) < 0\right] \\ &= P\left[\sum_{j=1}^{w_m}\left(|\,y_{0j}|^2 - |\,y_{1j}|^2\right) < 0\right] \end{aligned} \tag{14.7–3}$$

where w_m is the weight of the mth code word. But the probability in Equation 14.7–3 is just the probability of error for square-law combining of binary orthogonal FSK with w_mth-order diversity. That is,

$$P_2(m) = p^{w_m} \sum_{k=0}^{w_m-1} \binom{w_m - 1 + k}{k} (1-p)^k \tag{14.7–4}$$

$$\leq p^{w_m} \sum_{k=0}^{w_m-1} \binom{w_m - 1 + k}{k} = \binom{2w_m - 1}{w_m} p^{w_m} \tag{14.7–5}$$

where

$$p = \frac{1}{2 + \bar{\gamma}_c} = \frac{1}{2 + R_c \bar{\gamma}_b} \tag{14.7–6}$$

As an alternative, we may use the Chernov upper bound derived in Section 13.4, which in the present notation is

$$P_2(m) \leq [4p(1 - p)]^{w_m} \tag{14.7–7}$$

The sum of the binary error events over the $M - 1$ nonzero-weight code words gives an upper bound on the probability of error. Thus,

$$P_e \leq \sum_{m=2}^{M} P_2(m) \tag{14.7–8}$$

Since the minimum distance of the linear code is equal to the minimum weight, it follows that

$$(2 + R_c \bar{\gamma}_b)^{-w_m} \leq (2 + R_c \bar{\gamma}_b)^{-d_{\min}}$$

The use of this relation is conjunction with Equations 14.7–5 and 14.7–8 yields a simple, albeit looser, upper bound that may be expressed in the form

$$P_e < \frac{\sum_{m=2}^{M} \binom{2w_m - 1}{w_m}}{(2 + R_c \bar{\gamma}_b)^{d_{\min}}} \tag{14.7–9}$$

This simple bound indicates that the code provides an effective order of diversity equal to $d_{\min}$. An even simpler bound is the union bound

$$P_e < (M - 1)[4p(1 - p)]^{d_{\min}} \tag{14.7–10}$$

which is obtained from the Chernov bound given in Equation 14.7–7.

As an example serving to illustrate the benefits of coding for a Rayleigh fading channel, we have plotted in Figure 14.7–1 the performance obtained with the extended Golay (24,12) code and the performance of binary FSK and quaternary FSK each with dual diversity. Since the extended Golay code requires a total of 48 cells and $k = 12$, the bandwidth expansion factor $B_e = 4$. This is also the bandwidth expansion factor for binary and quaternary FSK with $L = 2$. Thus, the three types of waveforms are compared on the basis of the same bandwidth expansion factor. Note that at $P_b = 10^{-4}$, the Golay code outperforms quaternary FSK by more than 6 dB, and at $P_b = 10^{-5}$, the difference is approximately 10 dB.

The reason for the superior performanc of the Golay code is its large minimum distance ($d_{\min} = 8$), which translates into an equivalent eighth-order ($L = 8$) diversity. In contrast, the binary and quaternary FSK signals have only second-order diversity. Hence, the code makes more efficient use of the available channel bandwidth. The price that we must pay for the superior performance of the code is the increase in decoding complexity.

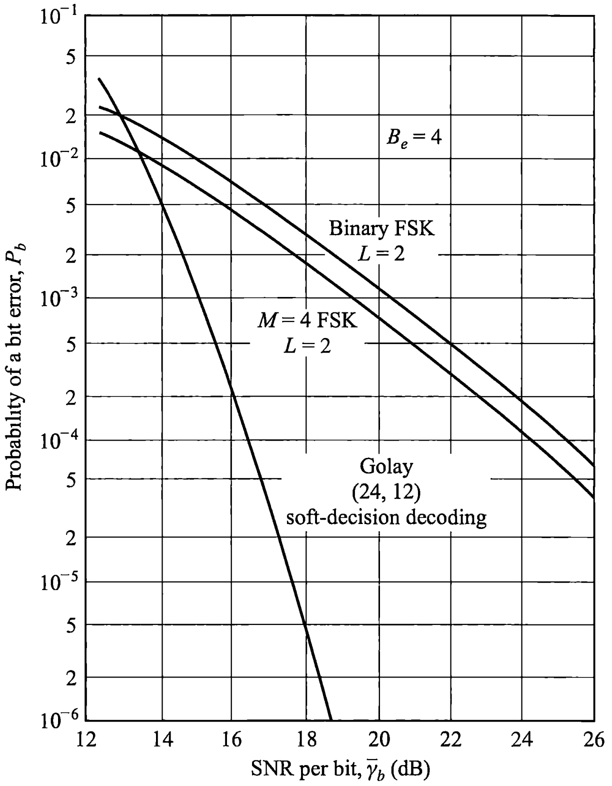


FIGURE 14.7–1
Example of performance obtained with conventional diversity versus coding for $B_e = 4$.

14.7–2 Probability of Error for Hard-Decision Decoding of Linear Block Codes

Bounds on the performance obtained with hard-decision decoding of a linear binary (n, k) code have already been given in Section 7.5–2. These bounds are applicable to a general binary-input, binary-output memoryless (binary symmetric) channel, and, hence, they apply without modification to a Rayleigh fading AWGN channel with statistically independent fading of the symbols in the code word. The probability of a bit error needed to evaluate these bounds when binary FSK with noncoherent detection is used as the modulation and demodulation technique is given by Equation 14.7–6.

A particularly interesting result is obtained when we use the Chernov upper bound on the error probability for hard-decision decoding given by

$$P_2(m) \le [4p(1-p)]^{w_m/2} \tag{14.7–11}$$

and P_e is upper-bounded by Equation 14.7–8. In comparison, the Chernov upper bound for $P_2(m)$ when soft-decision decoding is employed is given by Equation 14.7–7. We observe that the effect of hard-decision decoding is a reduction in the distance between any two code words by a factor of 2. When the minimum distance of a code is relatively small, the reduction of the distances by a factor of 2 is much more noticeable in a fading channel than in a nonfading channel.

For illustrative purposes we have plotted in Figure 14.7–2 the performance of the Golay (23, 12) code when hard-decision and soft-decision decoding are used. The difference in performance at $P_b = 10^{-5}$ is approximately 6 dB. This is a significant

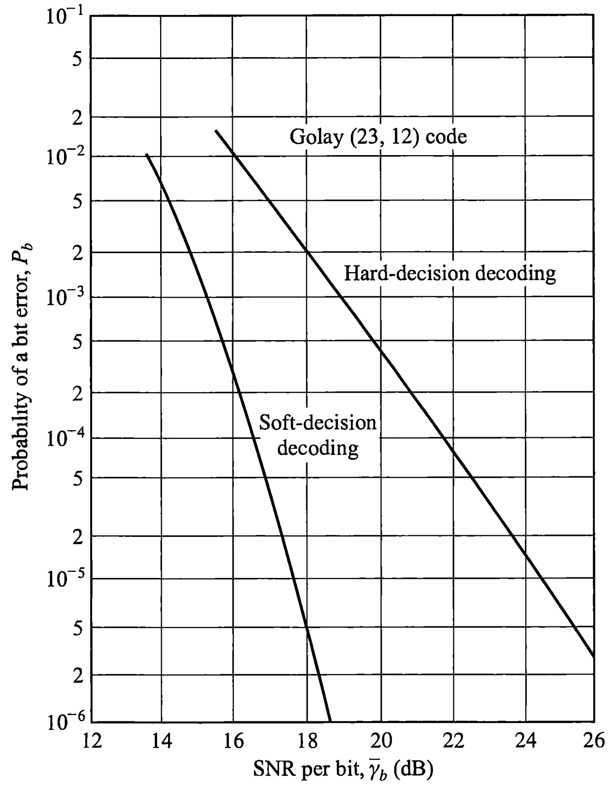


FIGURE 14.7–2
Comparison of performance between hard- and soft-decision decoding.

difference in performance compared with the 2-dB difference between soft- and hard-decision decoding in a nonfading AWGN channel. We also note that the difference in performance increases as P_b decreases. In short, these results indicate the benefits of soft-decision decoding over hard-decision decoding on a Rayleigh fading channel.

14.7–3 Upper Bounds on the Performance of Convolutional Codes for a Rayleigh Fading Channel

In this subsection, we derive the performance of binary convolutional codes when used on a Rayleigh fading AWGN channel. The encoder accepts k binary digits at a time and puts out n binary digits at a time. Thus, the code rate is $R_c = k/n$. The binary digits at the output of the encoder are transmitted over the Rayleigh fading channel by means of binary FSK, which is square-law-detected at the receiver. The decoder for either soft- or hard-decision decoding performs maximum-likelihood sequence estimation, which is efficiently implemented by means of the Viterbi algorithm.

First, we consider soft-decision decoding. In this case, the metrics computed in the Viterbi algorithm are simply sums of square-law-detected outputs from the demodulator. Suppose the all-zero sequence is transmitted. Following the procedure outlined in Section 8.2–2, it is easily shown that the probability of error in a pairwise comparison of the metric corresponding to the all-zero sequence with the metric corresponding to another sequence that merges for the first time at the all-zero state is

$$P_2(d) = p^d \sum_{k=0}^{d-1} \binom{d-1+k}{k} (1-p)^k \tag{14.7–12}$$

where d is the number of bit positions in which the two sequences differ and p is given by Equation 14.7–6. That is, $P_2(d)$ is just the probability of error for binary FSK with square-law detection and dth-order diversity. Alternatively, we may use the Chernov bound in Equation 14.7–7 for $P_2(d)$. In any case, the bit error probability is upper-bounded, as shown in Section 8.2–2 by the expression

$$p_b < \frac{1}{k} \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \tag{14.7–13}$$

where the weighting coefficients $\{\beta_d\}$ in the summation are obtained from the expansion of the first derivative of the transfer function $T(Y, Z)$, given by Equation 8.2–12.

When hard-decision decoding is performed at the receiver, the bounds on the error rate performance for binary convolutional codes derived in Section 8.2–2 apply. That is, P_b is again upper-bounded by the expression in Equation 14.7–13, where $P_2(d)$ is defined by Equation 8.2–16 for odd d and by Equation 8.2–17 for even d, or upper-bounded (Chernov bound) by Equation 8.2–15, and p is defined by Equation 14.7–6.

As in the case of block coding, when the respective Chernov bounds are used for $P_2(d)$ with hard-decision and soft-decision decoding, it is interesting to note that the effect of hard-decision decoding is to reduce the distances (diversity) by a factor of 2 relative to soft-decision decoding.

The following numerical results illustrate the error rate performance of binary, rate $1/n$, maximal free distance convolutional codes for $n = 2, 3$, and 4 with soft-decision Viterbi decoding. First of all, Figure 14.7–3 shows the performance of the rate $1/2$ convolutional codes for constraint lengths 3, 4, and 5. The bandwidth expansion factor for binary FSK modulation is $B_e = 2n$. Since an increase in the constraint length results in an increase in the complexity of the decoder to go along with the corresponding increase in the minimum free distance, the system designer can weight these two factors in the selection of the code.

Another way to increase the distance without increasing the constraint length of the code is to repeat each output bit m times. This is equivalent to reducing the code rate by a factor of m or expanding the bandwidth by the same factor. The result is a convolutional code that has a minimum free distance of md_{free}, where d_{free} is the minimum free distance of the original code without repetitions. Such a code is almost as good, from the viewpoint of minimum distance, as a maximum free distance, rate $1/mn$ code. The error rate performance with repetitions is upper-bounded by

$$P_b < \frac{1}{k} \sum_{d_{\text{free}}}^{\infty} \beta_d P_2(md) \tag{14.7–14}$$

where $P_2(md)$ is given by Equation 14.7–12. Figure 14.7–4 illustrates the performance of the rate $1/2$ codes with repetitions ($m = 1, 2, 3, 4$) for constraint length 5.

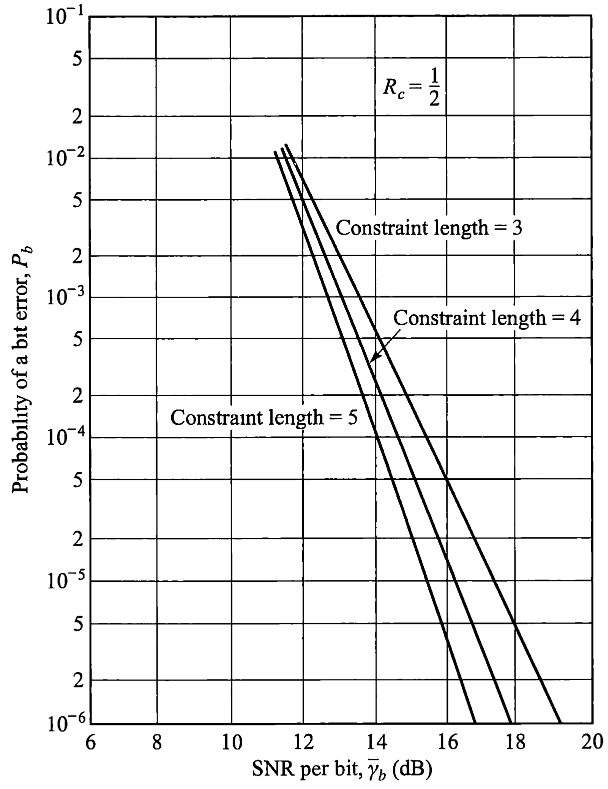


FIGURE 14.7–3
Performance of rate 1/2 binary convolutional codes with soft-decision decoding.

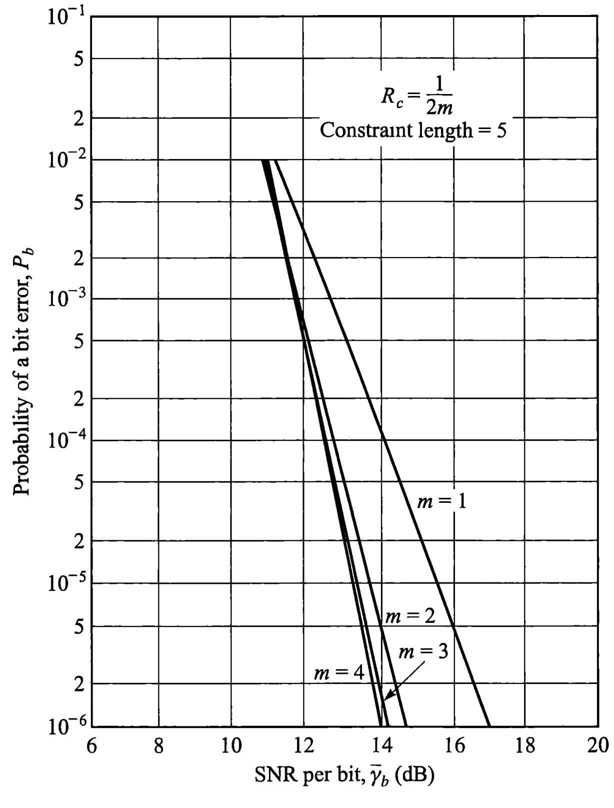


FIGURE 14.7–4
Performance of rate $1/2m$, constraint length 5, binary convolutional codes with soft-decision decoding.

14.7–4 Use of Constant-Weight Codes and Concatenated Codes for a Fading Channel

Our treatment of coding for a Rayleigh channel to this point was based on the use of binary FSK as the modulation technique for transmitting each of the binary digits in a code word. For this modulation technique, all the 2^k code words in the (n, k) code have identical transmitted energy. Furthermore, under the condition that the fading on the n transmitted tones is mutually statistically independent and identically distributed, the average received signal energy for the $M = 2^k$ possible code words is also identical. Consequently, in a soft-decision decoder, the decision is made in favor of the code word having the largest decision variable.

The condition that the received code words have identical average SNR has an important ramification in the implementation of the receiver. If the received code words do not have identical average SNR, the receiver must provide bias compensation for each received code word so as to render it equal energy. In general, the determination of the appropriate bias terms is difficult to implement because it requires the estimation of the average received signal power; hence, the equal-energy condition on the received code words considerably simplifies the receiver processing.

There is an alternative modulation method for generating equal-energy waveforms from code words when the code is constant-weight, i.e., when every code word has the same number of 1s. Note that such a code is non-linear. Nevertheless, suppose we assign a single tone or cell to each bit position of the 2^k code words. Thus, an (n, k) binary block code has n tones assigned. Waveforms are constructed by transmitting the tone corresponding to a particular bit in a code word if that bit is a 1; otherwise, that tone is not transmitted for the duration of the interval. This modulation technique for transmitting the coded bits is called *on–off keying* (OOK). Since the code is constant-weight, say, w, every coded waveform consists of w transmitted tones that depend on the positions of the 1s in each of the code words.

As in FSK, all tones in the OOK signal that are transmitted over the channel are assumed to fade independently across the frequency band and in time from one code word to another. The received signal envelope for each tone is described statistically by the Rayleigh distribution. Statistically independent additive white Gaussian noise is assumed to be present in each frequency cell.

The receiver employs maximum-likelihood (soft-decision) decoding to map the received waveform into one of the M possible transmitted code words. For this purpose, n matched filters are employed, each matched to one of the n frequency tones. For the assumed statistical independence of the signal fading for the n frequency cells and additive white Gaussian noise, the envelopes of the matched filter outputs are squared and combined to form the M decision variables

$$U_i = \sum_{j=1}^{n} c_{ij} |y_j|^2, \qquad i = 1, 2, \ldots, 2^k \tag{14.7–15}$$

where $|y_j|^2$ corresponds to the squared envelope of the filter corresponding to the jth frequency, where $j = 1, 2, \ldots, n$.

It may appear that the constant-weight condition severely restricts our choice of codes. This is not the case, however. To illustrate this point, we briefly describe some methods for constructing constant-weight codes. This discussion is by no means exhaustive.

Method 1: Non-linear transformation of a linear code In general, if in each word of an arbitrary binary code we substitute one binary sequence for every occurrence of a 0 and another sequence for each 1, a constant-weight binary block code will be obtained if the two substitution sequences are of equal weights and lengths. If the length of the sequence is ν and the original code is an (n, k) code, then the resulting constant-weight code will be an $(\nu n, k)$ code. The weight will be n times the weight of the substitution sequence, and the minimum distance will be the minimum distances of the original code times the distances between the two substitution sequences. Thus, the use of complementary sequences when ν is even results in a code with minimum distance $\nu d_{\min}$ and weight $\frac{1}{2}\nu n$.

The simplest form of this method is the case $\nu = 2$, in which every 0 is replaced by the pair 01 and every 1 is replaced by the complementary sequence 10 (or vice versa). As an example, we take as the initial code the (24,12) extended Golay code. The parameters of the original and the resultant constant-weight code are given in Table 14.7–1.

Note that this substitution process can be viewed as a separate encoding. This secondary encoding clearly does not alter the information content of a code word—it merely changes the form in which it is transmitted. Since the new code word is composed of pairs of bits—one "on" and one "off"—the use of OOK transmission of this code word produces a waveform that is identical to that obtained by binary FSK modulation for the underlying linear code.

Method 2: Expurgation In this method, we start with an arbitrary binary block code and select from it a subset consisting of all words of a certain weight. Several different constant-weight codes can be obtained from one initial code by varying the choice of the weight w. Since the code words of the resulting expurgated code can be viewed as a subset of all possible permutations of any one code word in the set, the term *binary expurgated permutation modulation* (BEXPERM) has been used by Gaarder (1971) to describe such a code. In fact, the constant-weight binary block codes constructed by the other methods may also be viewed as BEXPERM codes. This method

TABLE 14.7–1
Example of Constant-Weight Code Formed by Method 1

Code parameters	Original Golay	Constant-weight
n	24	48
k	12	12
M	4096	4096
$d_{\min}$	8	16
w	Variable	24

TABLE 14.7–2
Examples of Constant-Weight Codes Formed by Expurgation

Parameters	Original	Constant weight no. 1	Constant weight no. 2
n	24	24	24
k	12	9	11
M	4096	759	2576
$d_{\min}$	8	≥ 8	≥ 8
w	Variable	8	12

of generating constant-weight codes is in a sense opposite to the first method in that the word length n is held constant and the code size M is changed. The minimum distance for the constant-weight subset will clearly be no less than that of the original code. As an example, we consider the Golay (24, 12) code and form the two different constant-weight codes shown in Table 14.7–2.

Method 3: Hadamard matrices This method might appear to form a constant-weight binary block code directly, but it actually is a special case of the method of expurgation. In this method, a Hadamard matrix is formed as described in Section 7.3–5, and a constant-weight code is created by selection of rows (code words) from this matrix. Recall that a Hadamard matrix is an $n \times n$ matrix (n even integer) of 1s and 0s with the property that any row differs from any other row in exactly $\frac{1}{2}n$ positions. One row of the matrix is normally chosen as being all 0s.

In each of the other rows, half of the elements are 0s and the other half 1s. A Hadamard code of size $2(n-1)$ code words is obtained by selecting these $n-1$ rows and their complements. By selecting $M = 2^k \leq 2(n-1)$ of these code words, we obtain a Hadamard code, which we denote by $H(n, k)$, where each code word conveys k information bits. The resulting code has constant weight $\frac{1}{2}n$ and minimum distance $d_{\min} = \frac{1}{2}n$.

Since n frequency cells are used to transmit k information bits, the bandwidth expansion factor for the Hadamard $H(n, k)$ code is defined as

$$B_e = \frac{n}{k} \quad \text{cells per information bit}$$

which is simply the reciprocal of the code rate. Also, the average SNR per bit, denoted by $\bar{\gamma}_b$, is related to the average SNR per cell, $\bar{\gamma}_c$, by the expression

$$\bar{\gamma}_c = \frac{k}{\frac{1}{2}n}\bar{\gamma}_b = 2\frac{k}{n}\bar{\gamma}_b = 2R_c\bar{\gamma}_b = \frac{2\bar{\gamma}_b}{B_e} \tag{14.7–16}$$

Let us compare the performance of the constant-weight Hadamard codes under a fixed bandwidth constraint with a conventional M-ary orthogonal set of waveforms where each waveform has diversity L. The M orthogonal waveforms with diversity are equivalent to a block orthogonal code having a block length $n = LM$ and $k = \log_2 M$.

For example, if $M = 4$ and $L = 2$, the code words of the block orthogonal code are

$$\begin{aligned}
c_1 &= [1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0] \\
c_2 &= [0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0] \\
c_3 &= [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0] \\
c_4 &= [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1]
\end{aligned}$$

To transmit these code words using OOK modulation requires $n = 8$ cells, and since each code word conveys $k = 2$ bits of information, the bandwidth expansion factor $B_e = 4$. In general, we denote the block orthogonal code as $\mathrm{O}(n, k)$. The bandwidth expansion factor is

$$B_e = \frac{n}{k} = \frac{LM}{k} \tag{14.7–17}$$

Also, the SNR per bit is related to the SNR per cell by the expression

$$\bar{\gamma}_c = \frac{k}{L}\bar{\gamma}_b = M\left(\frac{k}{n}\right)\bar{\gamma}_b = M\frac{\bar{\gamma}_b}{B_e} \tag{14.7–18}$$

Now we turn our attention to the performance characteristics of these codes. First, the exact probability of a code word (symbol) error for M-ary orthogonal signaling over a Rayleigh fading channel with diversity was given in closed form in Section 13.4. As previously indicated, this expression is rather cumbersome to evaluate, especially if either L or M or both are large. Instead, we shall use a union bound that is very convenient. That is, for a set of M orthogonal waveforms, the probability of a symbol error can be upper-bounded as

$$\begin{aligned}
P_e &\le (M - 1)P_2(L) \\
&= (2^k - 1)P_2(L) < 2^k P_2(L)
\end{aligned} \tag{14.7–19}$$

where $P_2(L)$, the probability of error for two orthogonal waveforms, each with diversity L, is given by Equation 14.7–12 with $p = 1/(2 + \bar{\gamma}_c)$. The probability of bit error is obtained by multiplying P_e by $2^{k-1}/(2^k - 1)$, as explained previously.

A simple upper (union) bound on the probability of a code word error for the Hadamard $H(n, k)$ code is obtained by noting the probability of error in deciding between the transmitted code word and any other code word is bounded from above by $P_2\left(\frac{1}{2}d_{\min}\right)$, where $d_{\min}$ is the minimum distance of the code. Therefore, an upper bound on P_e is

$$P_e \le (M - 1)P_2\left(\tfrac{1}{2}d_{\min}\right) < 2^k P_2\left(\tfrac{1}{2}d_{\min}\right) \tag{14.7–20}$$

Thus the "effective order of diversity" of the code for OOK modulation is $\frac{1}{2}d_{\min}$. The bit error probability may be approximated as $\frac{1}{2}P_e$, or slightly overbounded by multiplying P_e by the factor $2^{k-1}/(2^k - 1)$, which is the factor used above for orthogonal codes. The latter was selected for the error probability computations given below.

Figure 14.7–5 illustrates the error rate performance of a selected number of Hadamard codes for several bandwidth expansion factors. The advantage resulting

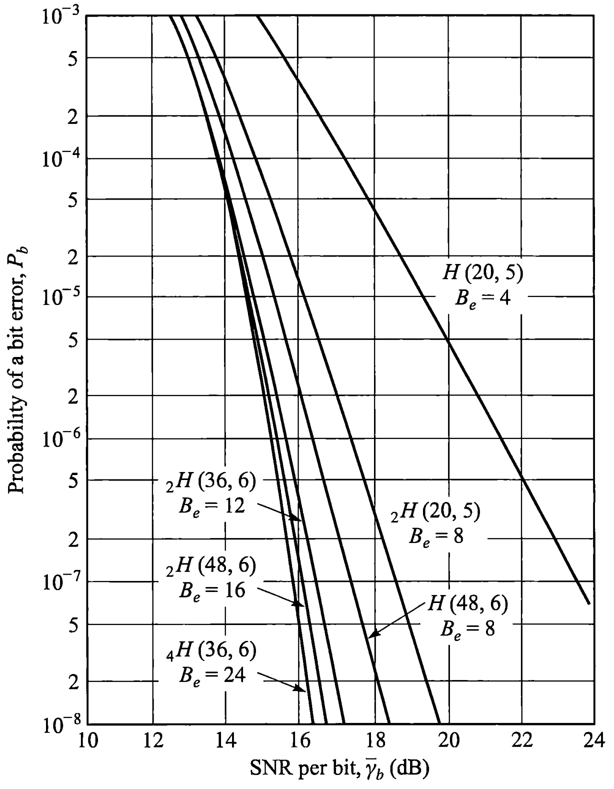


FIGURE 14.7–5
Performance of Hadamard codes.

from an increase in the size M of the alphabet (or k, since $k = \log_2 M$) and an increase in the bandwidth expansion factor is apparent from observation of these curves. Note, for example, that the $H(20, 5)$ code when repeated twice results in a code that is denoted by $_2H(20, 5)$ and has a bandwidth expansion factor $B_e = 8$. Figure 14.7–6 shows the performance of the Hadamard and block orthogonal codes compared on the basis of equal bandwidth expansion factors. It is observed that the error rate curves for the Hadamard codes are steeper than the corresponding curves for the block orthogonal codes. This characteristic behavior is due simply to the fact that, for the same bandwidth expansion factor, the Hadamard codes provide more diversity than block orthogonal codes. Alternatively, one may say that Hadamard codes provide better bandwidth efficiency than block orthogonal codes. It should be mentioned, however, that at low SNR, a lower-diversity code outperforms a higher-diversity code as a consequence of the fact that, on a Rayleigh fading channel, there is an optimum distribution of the total received SNR among the diversity signals. Therefore, the curves for the block orthogonal codes will cross over the curves for the Hadamard codes at the low-SNR (high-error-rate) region.

Method 4: Concatenation In this method, we begin with two codes: one binary and the other nonbinary. The binary code is the inner code and is an (n, k) constant-weight (non-linear) block code. The nonbinary code, which may be linear, is the outer code. To distinguish it from the inner code, we use uppercase letters, e.g., an (N, K) code, where N and K are measured in terms of symbols from a q-ary alphabet. The size q of the alphabet over which the outer code is defined cannot be greater than the

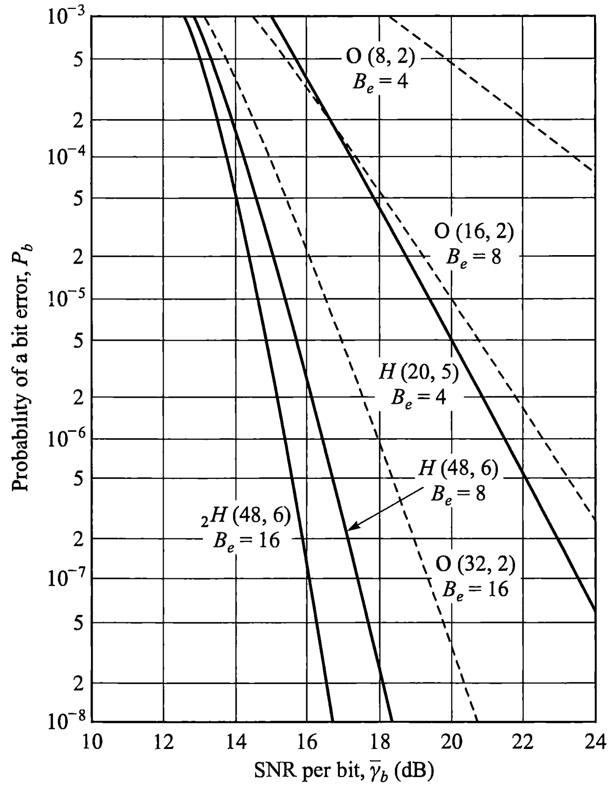


FIGURE 14.7–6
Comparison of performance between Hadamard codes and block orthogonal codes.

number of words in the inner code. The outer code, when defined in terms of the binary inner code words rather than q-ary symbols, is the new code.

An important special case is obtained when $q = 2^k$ and the inner code size is chosen to be 2^k. Then the number of words is $M = 2^{kK}$ and the concatenated structure is an (nN, kK) code. The bandwidth expansion factor of this concatenated code is the product of the bandwidth expansions for the inner and outer codes.

Now we shall demonstrate the performance advantages obtained on a Rayleigh fading channel by means of code concatenation. Specifically, we construct a concatenated code in which the outer code is a dual-k (nonbinary) convolutional code and the inner code is either a Hadamard code or a block orthogonal code. That is, we view the dual-k code with M-ary ($M = 2^k$) orthogonal signals for modulation as a concatenated code. In all cases to be considered, soft-decision demodulation and Viterbi decoding are assumed.

The error rate performance of the dual-k convolutional codes is obtained from the derivation of the transfer function given by Equation 8.7–2. For a rate-1/2, dual-k code with no repetitions, the bit error probability, appropriate for the case in which each k-bit output symbol from the dual-k encoder is mapped into one of $M = 2^k$ orthogonal code words, is upper-bounded as

$$P_b < \frac{2^{k-1}}{2^k - 1} \sum_{m=4}^{\infty} \beta_m P_2(m) \tag{14.7–21}$$

where $P_2(m)$ is given by Equation 14.7–12.

For example, a rate-1/2, dual-2 code may employ a 4-ary orthogonal code O(4, 2) as the inner code. The bandwidth expansion factor of the resulting concatenated code is, of course, the product of the bandwidth expansion factors of the inner and outer codes. Thus, in this example, the rate of the outer code is 1/2 and the inner code is 1/2. Hence, $B_e = (4/2)(2) = 4$.

Note that if every symbol of the dual-k is repeated r times, this is equivalent to using an orthogonal code with diversity $L = r$. If we select $r = 2$ in the example given above, the resulting orthogonal code is denoted as O(8, 2) and the bandwidth expansion factor for the rate-1/2, dual-2 code becomes $B_e = 8$. Consequently, the term $P_2(m)$ in Equation 14.7–21 must be replaced by $P_2(mL)$ when the orthogonal code has diversity L. Since a Hadamard code has an "effective diversity" $\frac{1}{2}d_{\min}$, it follows that when a Hadamard code is used as the inner code with a dual-k outer code, the upper bound on the bit error probability of the resulting concatenated code given by Equation 14.7–21 still applies if $P_2(m)$ is replaced by $P_2(\frac{1}{2}md_{\min})$. With these modifications, the upper bound on the bit error probability given by Equation 14.7–21 has been evaluated for rate-1/2, dual-k convolutional codes with either Hadamard codes or block orthogonal codes as inner codes. Thus the resulting concatenated code has a bandwidth exansion factor equal to twice the bandwidth expansion factor of the inner code.

First, we consider the performance gains due to code concatenation. Figure 14.7–7 illustrates the performance of dual-k codes with block orthogonal inner codes compared with the performance of block orthogonal codes for bandwidth expansion factors B_e = 4, 8, 16, and 32. The performance gains due to concatenation are very impressive.

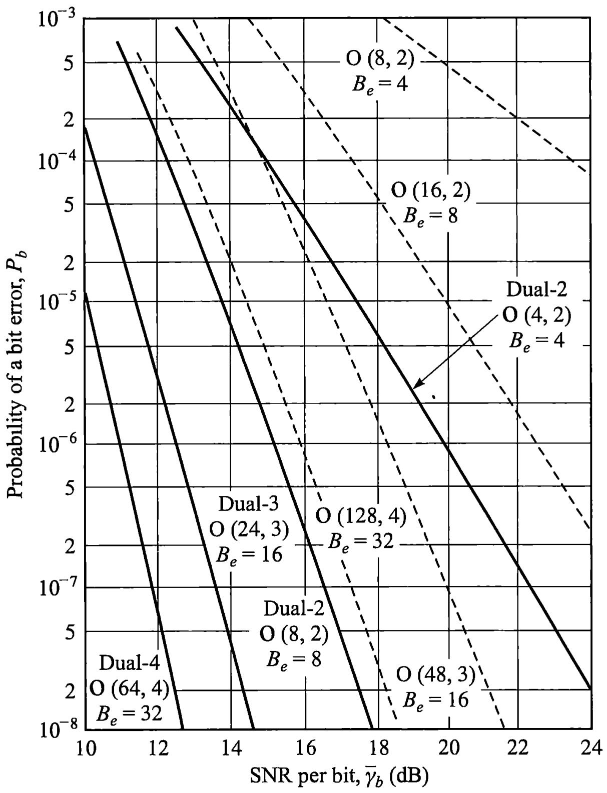


FIGURE 14.7–7
Comparison of performance between block orthogonal codes and dual-k with block orthogonal inner codes.

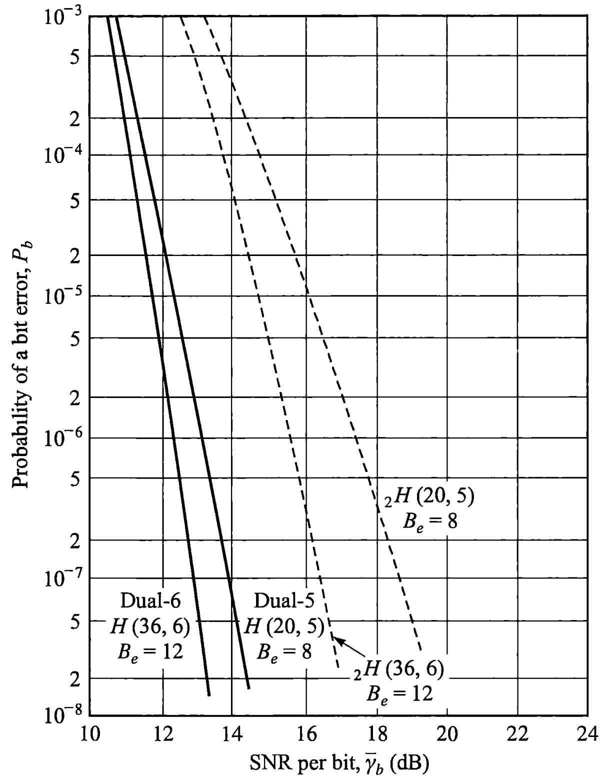


FIGURE 14.7–8
Comparison of performance between Hadamard codes and dual-k codes with Hadamard inner codes.

For example, at an error rate of 10^{-6} and $B_e = 8$, the dual-k code outperforms the orthogonal block code by 7.5 dB. In short, this gain may be attributed to the increased diversity (increase in minimum distance) obtained via code concatenation. Similarly, Figure 14.7–8 illustrates the performance of two dual-k codes with Hadamard inner codes compared with the performance of the Hadamard codes alone for $B_e = 8$ and 12. It is observed that the performance gains due to code concatenation are still significant, but certainly not as impressive as those illustrated in Figure 14.6–8. The reason is that the Hadamard codes alone yield a large diversity, so that the increased diversity arising from concatenation does not result in as large a gain in performance for the range of error rates covered in Figure 14.7–8.

The numerical results given above illustrate the performance advantages in using codes with good distance properties and soft-decision decoding on a Rayleigh fading channel as an alternative to conventional M-ary orthogonal signaling with diversity. In addition, the results illustrate the benefits of code concatenation on such a channel, using a dual-k convolutional code as the outer code and either a Hadamard code or a block orthogonal code as the inner code. Although dual-k codes were used for the outer code, similar results are obtained when a Reed–Solomon code is used for the outer code. There is an even greater choice in the selection of the inner code.

The important parameter in the selection of both the outer and the inner codes is the minimum distance of the resultant concatenated code required to achieve a specified level of performance. Since many codes will meet the performance requirements, the ultimate choice is made on the basis of decoding complexity and bandwidth requirements.

14.8
THE CHANNEL CUTOFF RATE FOR FADING CHANNELS

We studied the notion and significance of the channel cutoff rate for the general class of memoryless channels in Section 6.8. In the same section we obtained expressions for the channel cutoff rate for the special cases of a BSC channel and a binary-input, continuous-output Gaussian channel. In this section we extend those results to the case of fully interleaved Ricean and Rayleigh fading channels for the cases where CSI is available at the receiver.

We have seen in Section 6.8 that for a general memoryless channel the cutoff rate can be expressed by Equation 6.8–20 as

$$\begin{aligned} R_0 &= \max_{p(x)} \sup_{\lambda>0} \quad R_0(p,\lambda) \\ &= \max_{p(x)} \sup_{\lambda>0} \quad -\log_2 \left[\mathrm{E}\left[\Delta^{(\lambda)}_{X_1\to X_2} \right] \right] \end{aligned} \tag{14.8–1}$$

where for a symmetric channel model the maximum is achieved for $\lambda = \frac{1}{2}$, i.e., by substituting the Chernov bound by the Bhattacharyya bound, or substituting $\Delta^{(\lambda)}_{x_1\to x_2}$ by Δ_{x_1,x_2}. The values of $\Delta^{(\lambda)}_{x_1\to x_2}$ and Δ_{x_1,x_2} are given by Equation 6.8–10 as

$$\begin{aligned} \Delta^{(\lambda)}_{x_1\to x_2} &= \sum_{y\in\mathcal{Y}} p^{\lambda}(y|x_2)p^{1-\lambda}(y|x_1) \\ \Delta_{x_1,x_2} &= \sum_{y\in\mathcal{Y}} \sqrt{p(y|x_1)p(y|x_2)} \end{aligned} \tag{14.8–2}$$

where the summation on y corresponds to a discrete-output channel, which should be substituted by integration over the output space for a continuous-output channel. The expectation in Equation 14.8–1 is over all independent input distributions, i.e.,

$$\mathrm{E}\left[\Delta^{(\lambda)}_{X_1\to X_2}\right] = \left[\sum_{x_1\in\mathcal{X}}\sum_{x_2\in\mathcal{X}} p(x_1)p(x_2)\Delta^{(\lambda)}_{x_1\to x_2}\right] \tag{14.8–3}$$

where for continuous-input channels the summations are substituted by integrals.

14.8–1 Channel Cutoff Rate for Fully Interleaved Fading Channels with CSI at Receiver

For this channel model, ideal interleaving causes the channel model to be memoryless. The availability of CSI at the receiver can be interpreted as extending the channel output to be both the regular channel output y and the fading information. The channel is described as a memoryless model in which

$$y_i = r_i x_i + n_i \tag{14.8–4}$$

where r_i denotes the iid fading process and n_i is the iid noise process, which is assumed to be distributed according to $\mathcal{CN}(0, N_0)$ and is independent of the fading process. The channel inputs are assumed to be points in a complex constellation. For a Rayleigh fading channel the r_i's are iid drawn according to $\mathcal{CN}(0, 2\sigma^2)$. Since channel state information is available at the decoder, we can consider the pair (y_i, r_i) as the channel output. Therefore for this channel model P [output | input] can be written as

$$p(r, y|x) = p(r)p(y|r, x) \tag{14.8–5}$$

Since the channel model is symmetric, we use the Bhattacharyya bound and from Equation 14.8–2 we obtain

$$\begin{aligned}\Delta_{x_1,x_2} &= \int_0^\infty \left[\int_{-\infty}^\infty \sqrt{p(y|x_1, r)p(y|x_2, r)}\,dy\right] p(r)\,dr \\ &= \mathrm{E}\left[\int_{-\infty}^\infty \sqrt{p(y|x_1)p(y|x_1, r)}\,dy\right]\end{aligned} \tag{14.8–6}$$

where the expectation is taken with respect to the random variable R. For the channel model of Equation 14.8–4 we have

$$p(y|x, r) = \frac{1}{\pi N_0} e^{-\frac{|y-rx|^2}{N_0}} \tag{14.8–7}$$

Using Equation 14.8–7 after completing the square in the exponent and some manipulation, we obtain

$$\int_{-\infty}^\infty \sqrt{p(y|x_1)p(y|x_1, r)}\,dy = e^{-\frac{|r|^2}{4N_0}|x_1-x_2|^2} \tag{14.8–8}$$

or

$$\Delta_{x_1,x_2} = \mathrm{E}\left[e^{-\frac{|r|^2 d_{12}^2}{4N_0}}\right] \tag{14.8–9}$$

where $d_{12} = |x_1 - x_2|$. Defining

$$\alpha_{12} = \frac{d_{12}^2}{4N_0} \tag{14.8–10}$$

we obtain

$$\Delta_{x_1,x_2} = \mathrm{E}\left[e^{-\alpha_{12}|r|^2}\right] \tag{14.8–11}$$

In other words, Δ_{x_1,x_2} is equal to $\Theta_{|R|^2}(t)$, the moment generating function of the random variable $|R|^2$, i.e., the squared envelope of the fading process, when t is substituted with $-\alpha_{12}$.

For a Ricean fading channel $|R|$ has a Ricean distribution and $|R|^2$ has a noncentral χ^2 PDF with two degrees of freedom and parameters s and σ^2. From Table 2.3–3 we obtain the characteristic function of $|R|^2$, and from it we obtain

$$\Delta_{x_1,x_2} = \frac{1}{1 + 2\alpha_{12}\sigma^2} e^{-\frac{\alpha_{12}s^2}{1+2\alpha_{12}\sigma^2}} \tag{14.8–12}$$

By substituting the terms $A = s^2 + 2\sigma^2$ and $K = \frac{s^2}{2\sigma^2}$ in Equation 14.8–12, we have

$$\Delta_{x_1,x_2} = \frac{K+1}{K+1+A\alpha_{12}}\, e^{-\frac{AK\alpha_{12}}{K+1+A\alpha_{12}}} \tag{14.8–13}$$

Note that $A = \mathrm{E}[|R|^2]$ represents the average power gain of the channel. If we assume that $A = 1$, the transmitted and received powers become equal. For this case

$$\Delta_{x_1,x_2} = \frac{K+1}{K+1+\alpha_{12}}\, e^{-\frac{K\alpha_{12}}{K+1+\alpha_{12}}} \tag{14.8–14}$$

For a Rayleigh fading channel we have $s = K = 0$ and

$$\Delta_{x_1,x_2} = \frac{1}{1+\alpha_{12}} \tag{14.8–15}$$

Note that in all cases studied above, if $x_1 = x_2$, then $\alpha_{12} = 0$ and $\Delta_{12} = 1$.

For a BPSK modulation system the optimal $p(x)$ to achieve R_0 is a uniform distribution. To compute R_0, we need to find $\mathrm{E}\left[\Delta_{X_1,X_2}\right]$. For a uniform distribution on the inputs $\pm\sqrt{\mathcal{E}_s}$, the probability of $X_1 = X_2$ is $\frac{1}{2}$, and the probability of $X_1 \neq X_2$ is also $\frac{1}{2}$. For this latter case $d_{12}^2 = 4\mathcal{E}_s$, and from Equation 14.8–10 we obtain $\alpha_{12} = \mathcal{E}_s/N_0 = \mathrm{SNR}$. Therefore,

$$\mathrm{E}\left[\Delta_{X_1,X_2}\right] = \frac{1}{2} + \frac{1}{2}\Delta = \frac{\Delta+1}{2} \tag{14.8–16}$$

where

$$\Delta = \frac{K+1}{K+1+\mathrm{SNR}}\, e^{-\frac{K\mathrm{SNR}}{K+1+\mathrm{SNR}}} \tag{14.8–17}$$

and finally

$$\begin{aligned} R_0 &= -\log_2 \frac{\Delta+1}{2} \\ &= 1 - \log_2\left(1 + \frac{K+1}{K+1+\mathrm{SNR}}\, e^{-\frac{K\mathrm{SNR}}{K+1+\mathrm{SNR}}}\right) \end{aligned} \tag{14.8–18}$$

For the case of a Rayleigh fading channel, this relation reduces to

$$R_0 = 1 - \log_2\left(1 + \frac{1}{1+\mathrm{SNR}}\right) \tag{14.8–19}$$

For QPSK signaling the optimal input probability distribution is a uniform distribution. In this case, $d_{12}^2 = 0$, or $2\mathcal{E}_s$, or $4\mathcal{E}_s$ with probabilities $\frac{1}{4}$, $\frac{1}{2}$, and $\frac{1}{4}$, respectively. The corresponding values of α_{12} are 0, $\frac{\mathrm{SNR}}{2}$, and SNR, respectively. Substituting these values into Equation 14.8–14, we obtain

$$\mathrm{E}[\Delta] = \frac{1}{4} + \frac{1}{2}g\left(\frac{\mathrm{SNR}}{2}\right) + \frac{1}{4}g(\mathrm{SNR}) \tag{14.8–20}$$

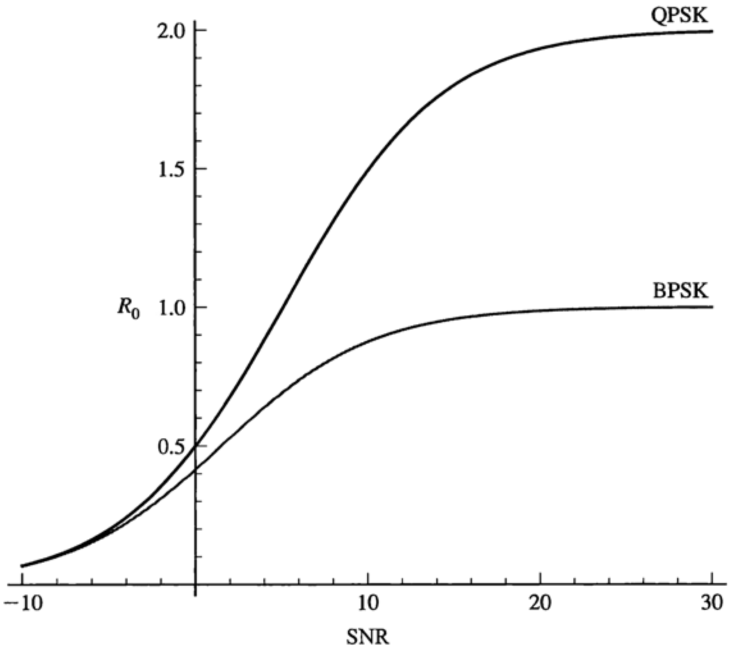


FIGURE 14.8–1
The cutoff rate versus SNR for BPSK and QPSK over a Rayleigh fading channel.

where

$$g(\alpha) = \frac{K+1}{K+1+\alpha}\, e^{-K\alpha/(K+1+\alpha)} \tag{14.8–21}$$

The Rayleigh fading case is obtained by putting $K = 0$ in Equation 14.8–21. The result is

$$\mathrm{E}\,[\Delta] = \frac{(\mathrm{SNR})^2 + 8\mathrm{SNR} + 8}{4(\mathrm{SNR}+2)(\mathrm{SNR}+1)} \tag{14.8–22}$$

Finally R_0 is obtained using

$$R_0 = -\log_2 \mathrm{E}\,[\Delta] \tag{14.8–23}$$

where $\mathrm{E}\,[\Delta]$ is obtained from Equations 14.8–20 and 14.8–22. Plots of R_0 versus $\mathrm{SNR} = \mathcal{E}_s/N_0$ for BPSK and QPSK in the case of a Rayleigh fading channel are shown in Figure 14.8–1.

14.9
BIBLIOGRAPHICAL NOTES AND REFERENCES

A comprehensive treatment of channel modeling, signaling, capacity issues, and coding techniques for fading channels can be found in Biglieri et al. (1998b). This paper summarizes and unifies the main results available on fading channel modeling, capacity, and coding up to 1998 and includes many references. Channel capacity for finite-state channels with different assumptions on the availability of state information are considered in Shannon (1958), Wolfowitz (1978), Salehi (1992), Cover and Chiang (2002), Goldsmith and Varaiya (1997), Goldsmith and Varaiya (1996), Abou-Faycal et al. (2001), and Ozarow et al. (1994).

Trellis-coded modulation for fading channels has been extensively treated in the books by Biglieri et al. (1991) and Jamali and Le-Ngoc (1994) as well as in the papers by Divsalar Simon (1988a, b, c), Sundberg and Seshadri (1993) and Salehi and Proakis (1995). Coding for fading channels is also the subject of the book by Biglieri (2005) where both coding and capacity issues under different assumptions have been treated. The book by **?**) also covers capacity and coding issues for wireless channels with emphasis on multiantenna systems.

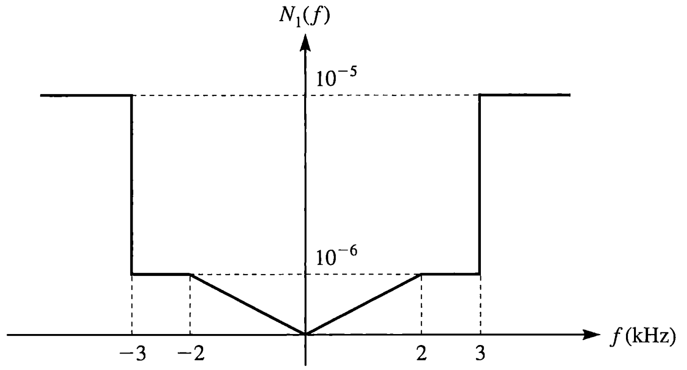
Bit-interleaved coded modulation introduced by Zehavi (1992) has been treated extensively in the paper by Caire et al. (1998). Other papers studying different aspects of this technique including error performance, iterative decoding, and optimal labeling under iterative decoding include the works of Ormeci et al. (2001), Martinez et al. (2006), and Li and Ritcey (1997, 1998, 1999).

The use of dual-*k* codes with M-ary orthogonal FSK was proposed in publications by Viterbi and Jacobs (1975) and Odenwalder (1976). The importance of coding for digital communications over a fading channel was also emphasized in a paper by Chase (1976). The benefits derived from concatenated coding with soft decision decoding for a fading channel were demonstrated by Pieper et al. (1978). The performance of dual-*k* codes with either block orthogonal codes or Hadamard codes as inner codes was investigated by Proakis and Rahman (1979). The error rate performance of maximal free-distance binary convolutional codes was evaluated by Rahman (1981).

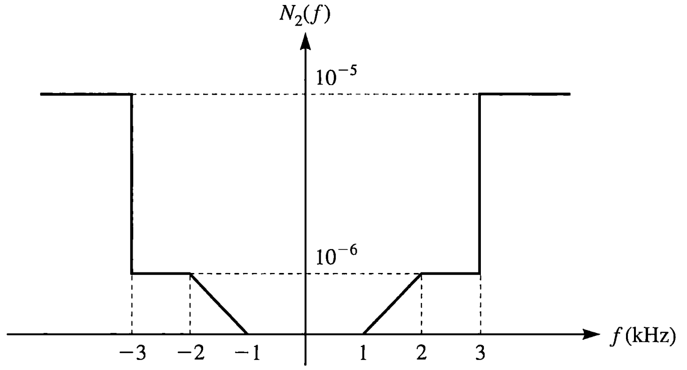
PROBLEMS

14.1 Channels 1 and 2 are both continuous-time additive Gaussian noise channels described by $Y_1(t) = X_1(t) + Z_1(t)$ and $Y_2(t) = X_2(t) + Z_2(t)$, respectively. $Z_1(t)$ and $Z_2(t)$ are the noise processes of the channels. It is assumed that $Z_1(t)$ and $Z_2(t)$ are zero-mean, *independent* Gaussian processes with power spectral densities $N_1(f)$ and $N_2(f)$ W/Hz, as shown in Figure P14.1. It is assumed that each channel has an input power constraint of 10 mW.

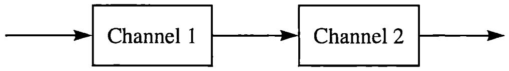
1. Determine C_1 and C_2, the capacities of the two channels (in bits per second).
2. If a binary memoryless source with $P(U = 0) = 1 - P(U = 1) = 0.4$ which generates 7500 symbols per second is to be transmitted once via channel 1 and once via channel 2, determine in each case the absolute minimum achievable error probability.
3. Now consider the two channel configurations shown in Figure P14.1. The first configuration is simply a concatenation of the two original channels. The second concatenation allows a processor with arbitrary complexity to be used between the two channels. In each case determine the absolute minimum achievable error probability for the binary source of part 2 when transmitted over the given channel configuration.
4. What is the capacity of channel 1 if the input power constraint is increased from 10 to 100 mW?



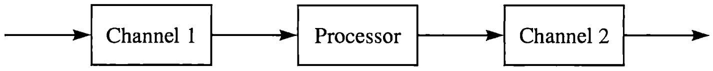
Noise spectral density in channel 1



Noise spectral density in channel 2



Configuration 1



Configuration 2

FIGURE P14.1

14.2 Consider the channel model shown in Figure 14.2–1 and assume both channel components are BSC channels with crossover probability $p = \frac{1}{2}$.

1. What is the ergodic capacity of this channel?
2. Now assume that the transmitter can control the state of the channel and the receiver has access to channel state information. What is the capacity of the resulting channel?

14.3 Using Equation 14.1–19, determine the capacity of a finite-state channel in which state information is only available at the receiver.

14.4 Using Equation 14.1–19, determine the capacity of a finite-state channel in which *the same* state information is available at the transmitter and the receiver.

14.5 Consider a BSC in which the channel can be in three states. In state $S = 0$ the output of the channel is always 0, regardless of the channel input. In state $S = 1$, the output is always 1, again regardless of the channel input. In state $S = 2$ the channel in noiseless, i.e., the output is always equal to the input. We assume that $P(S = 0) = P(S = 1) = \frac{p}{2}$.

1. Determine the capacity of this channel, assuming no state information is available to the transmitter or the receiver.
2. Determine the capacity of the channel, assuming that channel state information S is available at both sides.

14.6 In Problem 14.5 assume that the same noisy versions of state information are available at both sides; i.e., $Z = U = V$ is available where Z is a binary-valued random variable with

$$\mathrm{P}[Z = 0 \,|S = 0] = \mathrm{P}[Z = 1 \,|S = 1] = 1$$

$$\mathrm{P}[Z = 0 \,|S = 2] = \mathrm{P}[Z = 1 \,|S = 2] = \frac{1}{2}$$

Determine the capacity of this channel.

14.7 Consider the channel model shown in Figure 14.2–1. Assume that the top channel is a noiseless BSC channel for which crossover probability is zero and the bottom channel is a binary-input binary-output Z channel with $\mathrm{P}[Y = 1 \,|X = 1] = 1$ and $\mathrm{P}[Y = 0 \,|X = 0] = \frac{1}{2}$. The channel switches between the two states independently for each transmission, and the two states are equiprobable.

1. Determine the ergodic capacity of this channel when no state information is available.
2. Determine the ergodic capacity of the channel when perfect state information is available at both sides.
3. Determine the ergodic capacity of the channel when perfect state information is available at the receiver.

14.8 Prove that Equation 14.2–11 can be simplified in the form of Equation 14.2–13.

14.9 In Figure 14.4–1, determine the optimal rotation that maximizes the coding gain. What is the resulting coding gain?

14.10 A fading channel model that is flat in both time and frequency can be modeled as $\boldsymbol{y} = R\boldsymbol{x} + \boldsymbol{n}$, where the fading factor R remains constant for the entire duration of the transmission of the codeword. Determine the optimal decision rule for this channel for Ricean fading when the state information is available at the receiver and when it is not available.

14.11 The outage probability of a diversity combiner is defined as the percentage of time the instantaneous output SNR of the combiner is below some prescribed level for a specified number of diversity branches. Consider a communication system that employs multiple receiver antennas to achieve diversity in a Rayleigh fading channel. Suppose that selection diversity is used with N_r receiver antennas. If the average SNR is 20 dB, determine the probability that the instantaneous SNR drops below 10 dB when

1. $N_r = 1$
2. $N_r = 2$
3. $N_r = 4$

14.12 The Gauss-Markov model for a time-varying channel is given by

$$h(m+1) = \sqrt{1-\alpha}h(m) + \alpha w(m+1)$$

where $\{w(m)\}$ is a sequence of iid $\mathcal{CN}(0, 1)$ random variables independent of $h(0) \sim \mathcal{CN}(0, 1)$. The sampling time is T_s. The coherence time of this channel is controlled by the choice of parameter α.

1. Calculate the autocorrelation function of the sequence $\{h(m)\}$ denoted by $R_h(m)$.
2. Define coherence time as that corresponding to $R_h(m) = 0.5$. Determine the value of α in terms of T_s and the coherence time T_c.
3. Suppose that $\{h(m\}$ is transmitted from the receiver to the transmitter with a delay of T_s. The transmitter predicts the value of $h(m)$, say $\hat{h}(m)$, from the past samples $h(m-n)$ and $h(m-n-1)$. Thus

$$\hat{h}(m) = b_1 h(m-n) + b_2 h(m-n-1)$$

where the prediction coefficients b_1 and b_2 are determined to minimize the MSE

$$\mathrm{E}\left[|e|^2\right] = \mathrm{E}\left[|h(m) - \hat{h}(m)|^2\right]$$

Determine b_1 and b_2 that minimize MSE.

14.13 The rate 1/3, $K = 3$, binary convolutional code with transfer function given by Equation 8.1–21 is used for transmitting data over a Rayleigh fading channel via binary PSK.

1. Determine and plot the probability of error for hard decision decoding. Assume that the transmitted waveforms corresponding to the coded bits fade independently.
2. Determine and plot the probability of error for soft decision decoding. Assume that the waveforms corresponding to the coded bits fade independently.

14.14 Show that the pairwise error probability for a fully interleaved Rayleigh fading channel with fading process R_i can be bounded by

$$P_{\mathbf{x}\to\hat{\mathbf{x}}} \leq \prod_{i=1}^{n} \mathrm{E}\left[e^{-\frac{R_i^2|x_i-\hat{x}_i|^2}{4N_0}}\right]$$

where the expectation is taken with respect to R_i's. From above conclude the following bound on the pairwise error probability.

$$P_{\mathbf{x}\to\hat{\mathbf{x}}} \leq \prod_{i=1}^{n} \frac{1}{1 + |x_i - \hat{x}_i|^2/4N_0}$$

14.15 Determine the product distance and the free Euclidean distance of the coded modulation scheme shown in Figure 14.5–1.

14.16 Determine the product distance and the free Euclidean distance of the coded modulation scheme shown in Figure 14.5–2.

14.17 Show that the signal set assignment of Figure 14.5–5 provides a performance 1.315 dB superior to the signal set assignment of Figure 14.5–4 when used over an AWGN channel.

14.18 In Figure 14.6–3 show $\mathcal{X}_b^i$ for $b = 0, 1$ and for $1 \leq i \leq 4$ for both set partitioning labeling and Gray labeling.

15

Multiple-Antenna Systems

The use of multiple antennas at the receiver of a communication system is a standard method for achieving spatial diversity to combat fading without expanding the bandwidth of the transmitted signal. Spatial diversity can also be achieved by using multiple antennas at the transmitter. For example, it is possible to achieve dual diversity with two transmitting antennas and one receiving antenna, as we demonstrate in this chapter. We will also demonstrate that multiple transmitting antennas can be used to create multiple spatial channels and thus provide the capability to increase the data rate of a wireless communication system. This method is called *spatial multiplexing*.

15.1
CHANNEL MODELS FOR MULTIPLE-ANTENNA SYSTEMS

A communication system employing N_T transmitting antennas and N_R receiving antennas is generally called a *multiple-input, multiple-output (MIMO) system*, and the resulting spatial channel in such a system is called a *MIMO channel*. The special case in which $N_T = N_R = 1$ is called a *single-input, single-output (SISO) system*, and the corresponding channel is called a *SISO channel*. A second special case is one in which $N_T = 1$ and $N_R \geq 2$. The resulting system is called a *single-input, multiple-output (SIMO) system*, and the corresponding channel is called a *SIMO channel*. Finally, a third special case is one in which $N_T \geq 2$ and $N_R = 1$. The resulting system is called a *multiple-input, single-output (MISO) system*, and the corresponding channel is called a *MISO channel*.

In a MIMO system with N_T transmit antennas and N_R receive antennas, we denote the equivalent lowpass channel impulse response between the jth transmit antenna and the ith receive antenna as $h_{ij}(\tau; t)$, where τ is the age or delay variable and t is the time variable.† Thus, the randomly time-varying channel is characterized by the $N_R \times N_T$

---

†For convenience, the subscript on lowpass equivalent signals is omitted throughout this chapter.

matrix $\boldsymbol{H}(\tau; t)$, defined as

$$\boldsymbol{H}(\tau; t) = \begin{bmatrix} h_{11}(\tau; t) & h_{12}(\tau; t) & \cdots & h_{1N_T}(\tau; t) \\ h_{21}(\tau; t) & h_{22}(\tau; t) & \cdots & h_{2N_T}(\tau; t) \\ \vdots & \vdots & & \vdots \\ h_{N_R1}(\tau; t) & h_{N_R2}(\tau; t) & \cdots & h_{N_RN_T}(\tau; t) \end{bmatrix} \tag{15.1–1}$$

Suppose that the signal transmitted from the jth transmit antenna is $s_j(t)$, $j = 1, 2, \ldots, N_T$. Then the signal received at the ith antenna in the absence of noise may be expressed as

$$\begin{aligned} r_i(t) &= \sum_{j=1}^{N_T} \int_{-\infty}^{\infty} h_{ij}(\tau; t)\, s_j(t-\tau)\, d\tau \\ &= \sum_{j=1}^{N_T} h_{ij}(\tau; t) * s_j(\tau), \qquad i = 1, 2, \ldots, N_R \end{aligned} \tag{15.1–2}$$

where the asterisk denotes convolution. In matrix notation, Equation 15.1–2 is expressed as

$$\boldsymbol{r}(t) = \boldsymbol{H}(\tau; t) * \boldsymbol{s}(\tau) \tag{15.1–3}$$

where $\boldsymbol{s}(t)$ is an $N_T \times 1$ vector and $\boldsymbol{r}(t)$ is an $N_R \times 1$ vector.

For a frequency-nonselective channel, the channel matrix $\boldsymbol{H}$ is expressed as

$$\boldsymbol{H}(t) = \begin{bmatrix} h_{11}(t) & h_{12}(t) & \cdots & h_{1N_T}(t) \\ h_{21}(t) & h_{22}(t) & \cdots & h_{2N_T}(t) \\ \vdots & \vdots & & \vdots \\ h_{N_R1}(t) & h_{N_R2}(t) & \cdots & h_{N_RN_T}(t) \end{bmatrix} \tag{15.1–4}$$

In this case, the signal received at the ith antenna is simply

$$r_i(t) = \sum_{j=1}^{N_T} h_{ij}(t) s_j(t), \qquad i = 1, 2, \ldots, N_R \tag{15.1–5}$$

and, in matrix form, the received signal vector $\boldsymbol{r}(t)$ is given as

$$\boldsymbol{r}(t) = \boldsymbol{H}(t)\boldsymbol{s}(t) \tag{15.1–6}$$

Furthermore, if the time variations of the channel impulse response are very slow within a time interval $0 \le t \le T$, when T may be either the symbol interval or some general time interval, Equation 15.1–6 may be simply expressed as

$$\boldsymbol{r}(t) = \boldsymbol{H}\boldsymbol{s}(t), \qquad 0 \le t \le T \tag{15.1–7}$$

where $\boldsymbol{H}$ is constant within the time interval $0 \le t \le T$.

The slowly time-variant frequency-nonselective channel model embodied in Equation 15.1–7 is the simplest model for signal transmission in a MIMO channel. In the following two subsections, we employ this model to illustrate the performance characteristics of MIMO systems. At this point, we assume that the data to be transmitted are uncoded. Coding for MIMO channels is treated in Section 15.4.

15.1–1 Signal Transmission Through a Slow Fading Frequency-Nonselective MIMO Channel

Consider a wireless communication system that employs multiple transmitting and receiving antennas, as shown in Figure 15.1–1. We assume that there are N_T transmitting antennas and N_R receiving antennas. As illustrated in Figure 15.1–1, a block of N_T symbols is converted from serial to parallel, and each symbol is fed to one of N_T identical modulators, where each modulator is connected to a spatially separate antenna. Thus, the N_T symbols are transmitted in parallel and are received on N_R spatially separated receiving antennas.

In this section, we assume that each signal from a transmitting antenna to a receiving antenna undergoes frequency-nonselective Rayleigh fading. We also assume that the differences in propagation times of the signals from the N_T transmitting to the N_R receiving antennas are small relative to the symbol duration T, so that for all practical purposes, the signals from the N_T transmitting antennas to any receiving antenna are synchronous. Hence, we can represent the equivalent lowpass received signals at the receiving antennas in a signaling interval as

$$r_m(t) = \sum_{n=1}^{N_T} s_n h_{mn} g(t) + z_m(t), \qquad 0 \le t \le T, \qquad m = 1, 2, \ldots, N_R \quad (15.1\text{–}8)$$

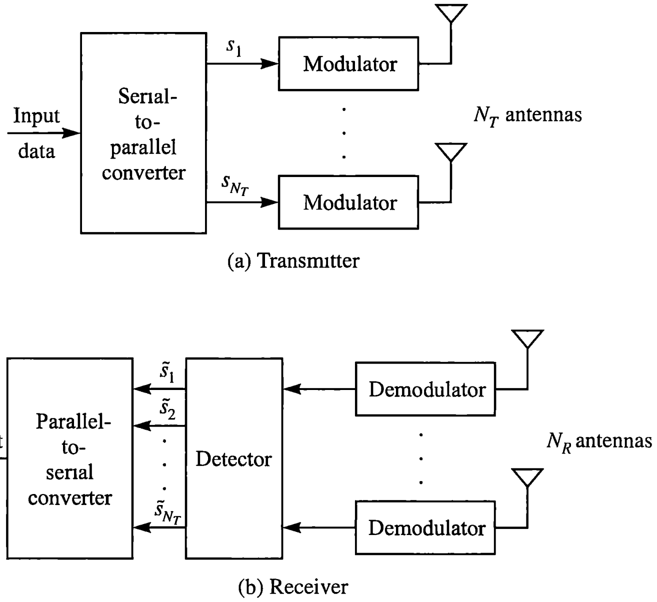

Input data
Serial-to-parallel converter
s_1
s_{N_T}
Modulator
Modulator
N_T antennas
(a) Transmitter
Output data
Parallel-to-serial converter
$\tilde{s}_1$
$\tilde{s}_2$
$\tilde{s}_{N_T}$
Detector
Demodulator
Demodulator
N_R antennas
(b) Receiver

FIGURE 15.1–1
A communication system with multiple transmitting and receiving antennas.

where $g(t)$ is the pulse shape (impulse response) of the modulation filters; h_{mn} is the complex-valued, circular zero-mean Gaussian channel gain between the nth transmitting antenna and the mth receiving antenna; s_n is the symbol transmitted on the nth antenna; and $z_m(t)$ is a sample function of an AWGN process. The channel gains $\{h_{mn}\}$ are identically distributed and statistically independent from channel to channel. The Gaussian sample functions $\{z_m(t)\}$ are identically distributed and mutually statistically independent, each having zero mean and two-sided power spectral density $2N_0$. The information symbols $\{s_n\}$ are drawn from either a binary or an M-ary PSK or QAM signal constellation.

The demodulator for the signal at each of the N_R receiving antennas consists of a matched filter to the pulse $g(t)$, whose output is sampled at the end of each symbol interval. The output of the demodulator corresponding to the mth receiving antenna can be represented as

$$y_m = \sum_{n=1}^{N_T} s_n h_{mn} + \eta_m, \qquad m = 1, 2, \ldots, N_R \tag{15.1–9}$$

where the energy of the signal pulse $g(t)$ is normalized to unity and η_m is the additive Gaussian noise component. The N_R soft outputs from the demodulators are passed to the signal detector. For mathematical convenience, Equation 15.1–9 may be expressed in matrix form as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{\eta} \tag{15.1–10}$$

where $\boldsymbol{y} = [y_1\, y_2\, \ldots\, y_{N_R}]^t$, $\boldsymbol{s} = [s_1\, s_2\, \ldots\, s_{N_T}]^t$, $\boldsymbol{\eta} = [\eta_1\, \eta_2\, \ldots\, \eta_{N_R}]^t$, and $\boldsymbol{H}$ is the $N_R \times N_T$ matrix of channel gains. Figure 15.1–2 illustrates the discrete-time model for the multiple transmitter and receiver signals in each signaling interval.

In the formulation of a MIMO system as described above, we observe that the transmitted symbols on the N_T transmitting antennas overlap totally in both time and frequency. As a consequence, there is interchannel interference in the signals $\{y_m, 1 \leq m \leq N_R\}$ received from the spatial channel. In the following subsection, we consider three different detectors for recovering the transmitted data symbols in a MIMO system.

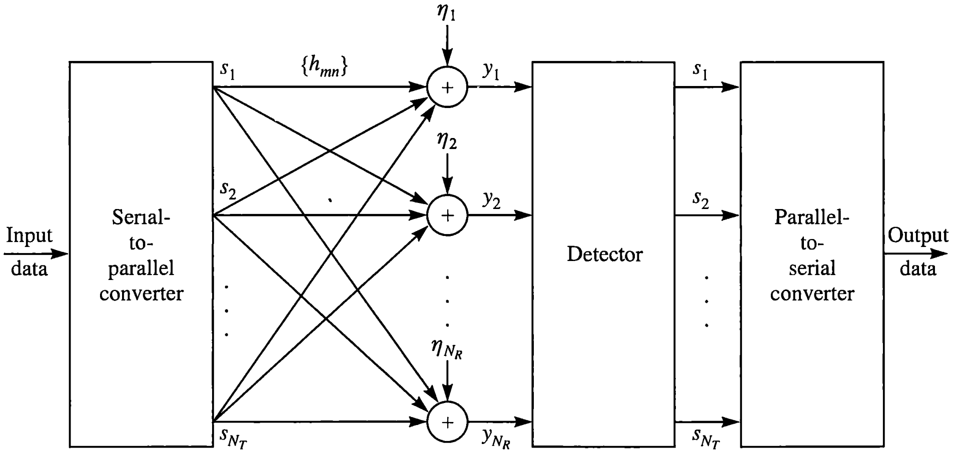


FIGURE 15.1–2

Discrete-time model of the communication system with multiple transmit and receive antennas in a frequency-nonselective slow fading channel.

15.1–2 Detection of Data Symbols in a MIMO System

Based on the frequency-nonselective MIMO channel model described in Section 15.1–1, we consider three different detectors for recovering the transmitted data symbols and evaluate their performance for Rayleigh fading and additive white Gaussian noise. Throughout this development, we assume that the detector knows the elements of the channel matrix $\boldsymbol{H}$ perfectly. In practice, the elements of $\boldsymbol{H}$ are estimated by using channel probe signals.

Maximum-Likelihood Detector (MLD) The MLD is the optimum detector in the sense that it minimizes the probability of error. Since the additive noise terms at the N_R receiving antennas are statistically independent and identically distributed (iid), zero-mean Gaussian, the joint conditional PDF $p(\boldsymbol{y}|\boldsymbol{s})$ is Gaussian. Therefore, the MLD selects the symbol vector $\hat{\boldsymbol{s}}$ that minimizes the Euclidean distance metric

$$\mu(\boldsymbol{s}) = \sum_{m=1}^{N_R} \left| y_m - \sum_{n=1}^{N_T} h_{mn} s_n \right|^2 \tag{15.1–11}$$

Minimum Mean-Square-Error (MMSE) Detector The MMSE detector linearly combines the received signals $\{y_m, 1 \le m \le N_R\}$ to form an estimate of the transmitted symbols $\{s_n, 1 \le n \le N_T\}$. The linear combining is represented in matrix form as

$$\hat{\boldsymbol{s}} = \boldsymbol{W}^H \boldsymbol{y} \tag{15.1–12}$$

where $\boldsymbol{W}$ is an $N_R \times N_T$ weighting matrix, which is selected to minimize the mean square error

$$J(\boldsymbol{W}) = E[\| \boldsymbol{e} \|^2] = E[\| \boldsymbol{s} - \boldsymbol{W}^H \boldsymbol{y} \|^2] \tag{15.1–13}$$

Minimization of $J(\boldsymbol{W})$ leads to the solution for the optimum weight vectors $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{N_T}$ as

$$\boldsymbol{w}_n = \boldsymbol{R}_{yy}^{-1} \boldsymbol{r}_{s_n y}, \qquad n = 1, 2, \ldots, N_T \tag{15.1–14}$$

where $\boldsymbol{R}_{yy} = E[\boldsymbol{y}\boldsymbol{y}^H] = \boldsymbol{H}\boldsymbol{R}_{ss}\boldsymbol{H}^H + N_0 \boldsymbol{I}$ is the $(N_R \times N_R)$ autocorrelation matrix of the received signal vector $\boldsymbol{y}$, $\boldsymbol{R}_{ss} = E[\boldsymbol{s}\boldsymbol{s}^H]$, $\boldsymbol{r}_{s_n y} = E[s_n^* \boldsymbol{y}]$, and $E[\boldsymbol{\eta}\boldsymbol{\eta}^H] = N_0 \boldsymbol{I}$. When the signal vector has uncorrelated, zero-mean components, $\boldsymbol{R}_{ss}$ is a diagonal matrix. Each component of the estimate $\hat{\boldsymbol{s}}$ is quantized to the closest transmitted symbol value.

Inverse Channel Detector (ICD) The ICD also forms an estimate of $\boldsymbol{s}$ by linearly combining the received signals $\{y_m, 1 \le m \le N_R\}$. In this case, if we set $N_T = N_R$, the weighting matrix $\boldsymbol{W}$ is selected so that the interchannel interference is completely eliminated, i.e., $\boldsymbol{W}^H = \boldsymbol{H}^{-1}$, hence

$$\begin{aligned} \hat{\boldsymbol{s}} &= \boldsymbol{H}^{-1} \boldsymbol{y} \\ &= \boldsymbol{s} + \boldsymbol{H}^{-1} \boldsymbol{\eta} \end{aligned} \tag{15.1–15}$$

Each element of the estimate $\hat{\boldsymbol{s}}$ is then quantized to the closest transmitted symbol value. We note that the ICD estimate $\hat{\boldsymbol{s}}$ is not corrupted by interchannel interference.

However, this also implies that the ICD does not exploit the signal diversity inherent in the received signal, as we will observe below.

When $N_R > N_T$, the weighting matrix $\boldsymbol{W}$ may be selected as the pseudoinverse of the channel matrix, i.e.,

$$\boldsymbol{W}^H = (\boldsymbol{H}^H\boldsymbol{H})^{-1}\boldsymbol{H}^H$$

Error Rate Performance of the Detectors The error rate performance of the three detectors in a Rayleigh fading channel is most easily assessed by computer simulation of the MIMO system. Figures 15.1–3 and 15.1–4 illustrate the binary error rate (BER) for binary PSK modulation with $(N_T, N_R) = (2, 2)$ and $(N_T, N_R) = (2, 3)$, respectively. In both cases, the variances of the channel gains are identical, and their sum is normalized to unity, i.e.,

$$\sum_{n,m} E\left[|h_{mn}|^2\right] = 1 \tag{15.1–16}$$

The BER for binary PSK modulation is plotted as a function of the average SNR per bit. With the normalization of the variances in the channel gains $\{h_{mn}\}$ as given by Equation 15.1–16, the average received energy is simply the transmitted signal energy per symbol.

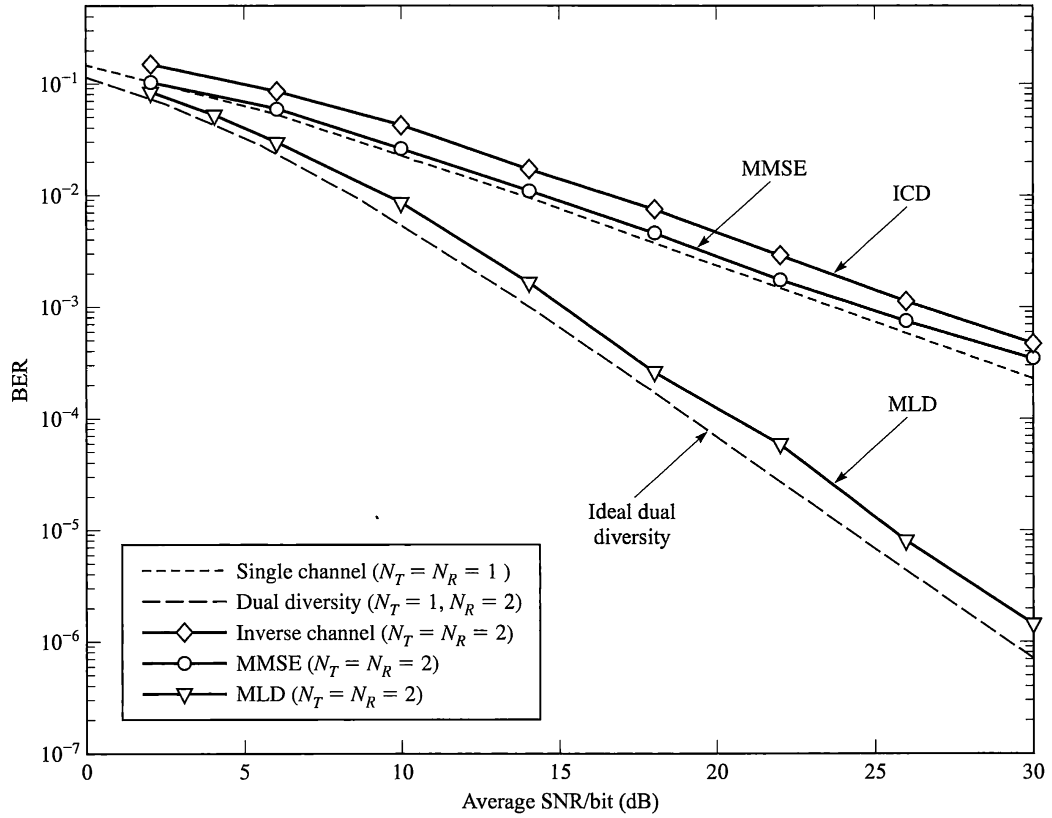


FIGURE 15.1–3
Performance of MLD, MMSE, and inverse channel detectors with $N_R = 2$ receiving antennas.

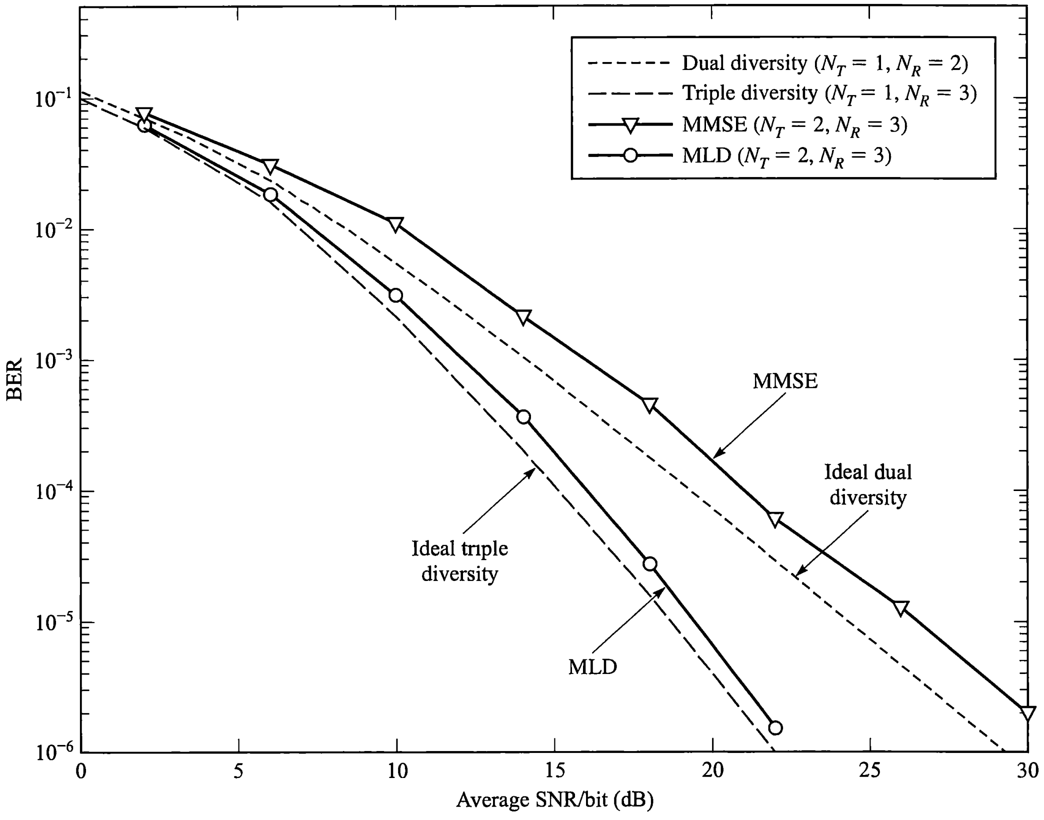


FIGURE 15.1–4
Performance of MLD and MMSE detectors with $N_R = 3$ receiving antennas.

The performance results in Figures 15.1–3 and 15.1–4 illustrate that the MLD exploits the full diversity of order N_R available in the received signal, and thus its performance is comparable to that of a maximal ratio combiner (MRC) of the N_R received signals, without the presence of interchannel interference, i.e., $(N_T, N_R) = (1, N_R)$. The two linear detectors—the MMSE detector and the ICD—achieve an error rate that decreases inversely as the SNR raised to the $(N_R - 1)$ power for $N_T = 2$ transmitting antennas. Thus, when $N_R = 2$, the two linear detectors achieve no diversity, and when $N_R = 3$, the linear detectors achieve dual diversity. We also note that the MMSE detector outperforms the ICD, although both achieve the same order of diversity. In general, with spatial multiplexing (N_T antennas transmitting independent data streams), the MLD detector achieves a diversity of order N_R, and the linear detectors achieve a diversity of order $N_R - N_T + 1$, for any $N_R \geq N_T$. In effect, with N_T antennas transmitting independent data streams and N_R receiving antennas, a linear detector has N_R degrees of freedom. In detecting any one data stream, in the presence of $N_T - 1$ interfering signals from the other transmitting antennas, the linear detectors utilize $N_T - 1$ degrees of freedom to cancel the $N_T - 1$ interfering signals. Therefore, the effective order of diversity for the linear detectors is $N_R - (N_T - 1) = N_R - N_T + 1$.

Let us now compare the computational complexity of the three detectors. We observe that the complexity of the MLD grows exponentially as M^{N_T}, where M is the number of points in the signal constellation, whereas the linear detectors have a complexity that grows linearly with N_T and N_R. Therefore, the computational complexity of the MLD is significantly larger when M and N_T are large. However, for a small number of transmitting antennas and signal points, say $N_T \leq 4$ and $M = 4$, the computational complexity of the MLD is not excessive.

Other Detector Structures and Algorithms

As we have observed, the MLD is the optimum detector, hence, it minimizes the symbol error probability. The two linear detectors, the ICD and the MMSE detector, are suboptimum in terms of performance, but have low computational complexity. Another class of detectors is nonlinear detectors whose performance is generally better than that of linear detectors, but their computational complexity is greater.

An example of a nonlinear detector is one that employs successive cancellation of symbols from the received signal once the symbols are detected. One method for accomplishing symbol cancellation is to employ the ICD or MMSE detector on the first pass through the data. From the linearly detected symbols, we select the symbol having the highest SNR, i.e., which is the most reliable. This symbol can be multiplied by the appropriate row of the channel matrix $\boldsymbol{H}$ and the result subtracted from the received signals, leaving us with a received signal containing $N_T - 1$ symbols. Then we repeat the detection procedure for the received signal containing the $N_T - 1$ symbols. Thus, N_T iterations are employed to detect the N_T transmitted symbols. This successive cancellation technique, applied to a MIMO system, is essentially a multiuser detection method that is further treated in Chapter 16.

This is just one example of a nonlinear detection algorithm that may be employed to detect the data. Such schemes have greater computational complexity than the linear detectors described, but their performance is generally better.

Another suboptimum detection method that is simpler to implement than MLD is *sphere detection* (also called *sphere decoding*). In sphere detection, the search for the most probable transmitted signal vector $\boldsymbol{s}$ is limited to a set of points $\boldsymbol{Hs}$ that lie within an N_R-dimensional hypersphere of fixed radius centered on the received signal vector $\boldsymbol{y}$. Thus, compared with MLD in which the search for the most probable signal vector $\boldsymbol{s}$ encompasses all possible points $\boldsymbol{Hs}$, sphere detection involves a search over a limited set of received signal points. Consequently, the computational complexity is decreased at a cost of an increase in the error probability. Clearly, as the radius of the sphere is increased, the performance of the sphere detector approaches the performance of the MLD. Computationally efficient algorithms for sphere detection, i.e., determining the signal points $\boldsymbol{Hs}$ that lie inside a sphere of a given radius centered on the received vector $\boldsymbol{y}$, have been published by Fincke and Pohst (1985), Viterbo and Boutros (1999), Damen et al. (2000), deJong and Willink (2002), and Hochwald and ten Brink (2003).

Another nonlinear method that exploits the signal diversity inherent in the received signal vector $\boldsymbol{y}$ and provides near MLD performance is based on lattice reduction. For example, recall that if the elements of the n-dimensional signal vector $\boldsymbol{s}$ are taken from a square QAM signal constellation, the set of signal vectors can be viewed as a subset of an n-dimensional lattice. Hence, the noiseless received signal vector $\boldsymbol{Hs}$ is a subset of a lattice that is transformed (distorted) by the channel matrix $\boldsymbol{H}$. The basis vectors for this transformed lattice are the columns of the matrix $\boldsymbol{H}$, which, in general, are not orthogonal. However, the basis vectors of the transformed lattice may be orthogonalized and reduced in magnitude, resulting in a new generator matrix $\boldsymbol{B}$ that is related to $\boldsymbol{H}$ through the transformation $\boldsymbol{B} = \boldsymbol{H}\boldsymbol{F}$, where the columns of $\boldsymbol{B}$ are orthogonal and $\boldsymbol{F}$ is a unimodular matrix with elements having integer real and imaginary components, such that $\boldsymbol{F}$ satisfies the condition $\det(\boldsymbol{F}) = \pm 1$ or $\pm j$. The inverse $\boldsymbol{F}^{-1}$ of such a matrix always exists.

We may use this basis transformation to express the received signal vector $\boldsymbol{y}$ as

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{H}\boldsymbol{s} + \boldsymbol{\eta} \\ &= (\boldsymbol{B}\boldsymbol{F}^{-1})\boldsymbol{s} + \boldsymbol{\eta} \end{aligned}$$

We define the vector $\boldsymbol{w}$ as $\boldsymbol{w} = \boldsymbol{F}^{-1}\boldsymbol{s}$, so that $\boldsymbol{y}$ may be expressed as

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{B}\boldsymbol{w} + \boldsymbol{\eta} \\ &= (\boldsymbol{H}\boldsymbol{F})\boldsymbol{w} + \boldsymbol{\eta} \end{aligned}$$

Now, the ICD may be applied to detect the transformed signal vector $\boldsymbol{w}$ by inverting $\boldsymbol{B}$ and making hard decisions on the resulting elements of the vector $\boldsymbol{B}^{-1}\boldsymbol{y}$ to yield the vector $\hat{\boldsymbol{w}}$. An estimate of the signal vector $\boldsymbol{s}$ is obtained by the linear transformation $\hat{\boldsymbol{s}} = \boldsymbol{F}\hat{\boldsymbol{w}}$. This detection method has been shown to yield an order of diversity comparable to MLD (for reference, see Yao and Wornell (2002)). Further discussion on lattice reduction is given in Section 16.4–4, in the context of MIMO broadcast channels.

Signal Detection When Channel Is Known at the Transmitter and Receiver

The MLD, MMSE, and ICD techniques are based on knowing the channel matrix $\boldsymbol{H}$ at the receiver. Another linear processing technique may be devised when the channel matrix $\boldsymbol{H}$ is known at the transmitter as well as the receiver. In this method, the singular value decomposition (SVD) of the channel matrix $\boldsymbol{H}$, assumed to be of rank r, may be expressed as

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H \tag{15.1–17}$$

where $\boldsymbol{U}$ is an $N_R \times r$ matrix, $\boldsymbol{V}$ is an $N_T \times r$ matrix, and $\boldsymbol{\Sigma}$ is an $r \times r$ diagonal matrix with diagonal elements the singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$ of the channel. The column vectors of the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthonormal. Hence $\boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{I}_r$ and $\boldsymbol{V}^H\boldsymbol{V} = \boldsymbol{I}_r$, where $\boldsymbol{I}_r$ is the $r \times r$ identity matrix. If we process an $r \times 1$ signal vector $\boldsymbol{s}$ at the transmitter by the linear transformation

$$\boldsymbol{s}_v = \boldsymbol{V}\boldsymbol{s} \tag{15.1–18}$$

then the received signal vector $\boldsymbol{y}$ is

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s}_v + \boldsymbol{\eta} = \boldsymbol{H}\boldsymbol{V}\boldsymbol{s} + \boldsymbol{\eta} \tag{15.1–19}$$

At the receiver, we process the received signal vector $\boldsymbol{y}$ by the linear transformation $\boldsymbol{U}^H$. Thus,

$$\begin{aligned} \hat{\boldsymbol{s}} &= \boldsymbol{U}^H\boldsymbol{y} = \boldsymbol{U}^H\boldsymbol{H}\boldsymbol{V}\boldsymbol{s} + \boldsymbol{U}^H\boldsymbol{\eta} \\ &= \boldsymbol{U}^H\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H\boldsymbol{V}\boldsymbol{s} + \boldsymbol{U}^H\boldsymbol{\eta} = \boldsymbol{\Sigma}\boldsymbol{s} + \boldsymbol{U}^H\boldsymbol{\eta} \end{aligned} \tag{15.1–20}$$

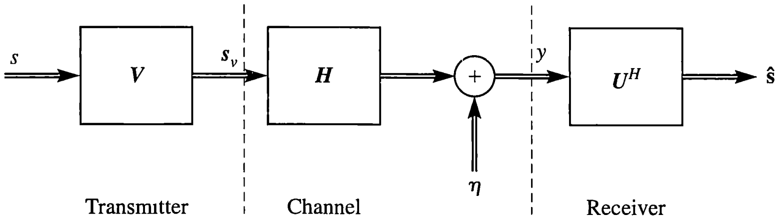


FIGURE 15.1–5
Signal processing and detection in a MIMO system when the channel is known at the transmitter and the receiver.

Therefore, the elements of the received signal are decouptled and may be detected individually. The scaling of the transmitted symbols by the singular values $\{\sigma_i\}$ may be compensated either at the transmitter by using the linear transformation $V\Sigma^{-1}$ in place of V or at the receiver by the linear transformation $\Sigma^{-1}U^H$. A block diagram of the MIMO communication system is illustrated in Figure 15.1–5.

From the expression for the estimate of the signal vector s given by Equation 15.1–20 we observe that the SVD method does not exploit the signal diversity provided by the channel. This is the main disadvantage in decoupling the received signal vector y by means of the SVD.

15.1–3 Signal Transmission Through a Slow Fading Frequency-Selective MIMO Channel

In this section we consider transmission through a frequency-selective MIMO channel in which the time variations of the impulse responses $\{h_{ij}(\tau; t)\}$ are very slow compared to the symbol rate $1/T$. According to Equations 15.1–2 and 15.1–3, the signal received from the frequency-selective MIMO channel may be expressed as

$$r_i(t) = \sum_{j=1}^{N_T} \int_{-\infty}^{\infty} h_{ij}(\tau; t) s_j(t-\tau)\, d\tau + z_i(t), \qquad i = 1, 2, \ldots, N_R \qquad (15.1\text{–}21)$$

where $z_i(t)$ represents the additive noise at the ith receive antenna. Let the signal transmitted in the nth signal interval be $s_j(t) = s_j(n)g(t-nT)$, where $g(t)$ is the impulse response of the modulation filters and $\{s_j(n)\}$ is the set of N_T information symbols. After substituting for $s_j(t)$ in Equation 15.1–21, we obtain

$$r_i(t) = \sum_n \sum_{j=1}^{N_T} s_j(n) \int_{-\infty}^{\infty} h_{ij}(\tau; t) g(t-nT-\tau)\, d\tau + z_i(t), \qquad i = 1, 2, \ldots, N_R \qquad (15.1\text{–}22)$$

It is convenient to process the received signal in sampled form. Consequently, we may sample the received signal $r_i(t)$ at some suitable sampling rate $F_s = J/T$, where J is a positive integer. For example, we may select $J = 2$, so that there are two samples per symbol. Such a sampling rate is appropriate when the impulse response $g(t)$ of the modulation filters is band-limited to $|f| \le 1/T$.

At each antenna, the received signal is passed through a bank of N_T finite-duration impulse response (FIR) filters, where each filter spans K samples. The filter coefficients at time instant n are denoted as $\{a_{ij}(k; n), k = 0, 1, \ldots, K\}$ and are assumed to be complex-valued in general. Suppose that these FIR filters function as linear equalizers. Then the outputs of the FIR filters from the N_R receive antennas may be used to form estimates of the transmitted information symbols. Thus, the estimate of the jth information symbol transmitted at time instant n may be expressed as

$$\hat{s}_j(n) = \sum_{i=1}^{N_R} \left[\sum_{k=0}^{K-1} a_{ij}(k; n) r_i(n-k) \right], \qquad j = 1, 2, \ldots, N_T \tag{15.1–23}$$

where $\hat{s}_j(n)$ denotes the estimate of $s_j(n)$.

The estimates given by Equation 15.1–23 can be expressed more compactly in matrix form as

$$\hat{\boldsymbol{s}}(n) = \boldsymbol{A}^H(n)\boldsymbol{r}(n) \tag{15.1–24}$$

where the matrix $\boldsymbol{A}(n)$ and the vector $\boldsymbol{r}(n)$ are defined as

$$\boldsymbol{A}(n) = \begin{bmatrix} \boldsymbol{a}_{11}^*(n) & \boldsymbol{a}_{12}^*(n) & \cdots & \boldsymbol{a}_{1N_T}^*(n) \\ \boldsymbol{a}_{21}^*(n) & \boldsymbol{a}_{22}^*(n) & \cdots & \boldsymbol{a}_{2N_T}^*(n) \\ \vdots & \vdots & & \\ \boldsymbol{a}_{N_R1}^*(n) & \boldsymbol{a}_{N_R2}^*(n) & \cdots & \boldsymbol{a}_{N_RN_T}^*(n) \end{bmatrix} \tag{15.1–25}$$

$$\boldsymbol{r}(n) = \begin{bmatrix} \boldsymbol{r}_1(n) \\ \boldsymbol{r}_2(n) \\ \vdots \\ \boldsymbol{r}_{N_R}(n) \end{bmatrix}$$

where $\{\boldsymbol{a}_{ij}(n)\}$ and $\{\boldsymbol{r}_j(n)\}$ are column vectors of dimension K and $\boldsymbol{A}^H(n) = [\boldsymbol{A}(n)]^H = [\boldsymbol{a}_{ij}^*(n)]^H = [\boldsymbol{a}_{ji}^t(n)]$. Figure 15.1–6 illustrates the structure of the demodulator for $N_T = 2$ transmitting antennas and $N_R = 3$ receiving antennas.

The estimate $\hat{\boldsymbol{s}}(n)$ is fed to the detector which compares each element of $\hat{\boldsymbol{s}}(n)$ with the possible transmitted symbols and selects the symbol $s_j(n)$ that is closest in Euclidean distance to $\hat{s}_j(n)$.

When the channel impulse responses $\{h_{ij}(\tau; t)\}$ change slowly with time, the coefficients of the FIR equalizers can be adjusted adaptively to minimize the mean square error (MSE) between the desired data symbols $\{s_j(n), j = 1, 2, \ldots, N_T\}$ and the estimates $\{\hat{s}_j(n), j = 1, 2, \ldots, N_T\}$. Initial adjustment of the coefficients $\{\boldsymbol{a}_{ij}(n)\}$ may be accomplished by transmitting a finite-duration sequence of training symbol vectors from the N_T transmit antennas. In the training mode, the error signal is formed as

$$\begin{aligned} \boldsymbol{e}(n) &= \boldsymbol{s}(n) - \hat{\boldsymbol{s}}(n) \\ &= \boldsymbol{s}(n) - \boldsymbol{A}^H(n)\boldsymbol{r}(n) \end{aligned} \tag{15.1–26}$$

or, equivalently, as

$$e_j(n) = s_j(n) - \hat{s}_j(n), \qquad j = 1, 2, \ldots, N_T \tag{15.1–27}$$

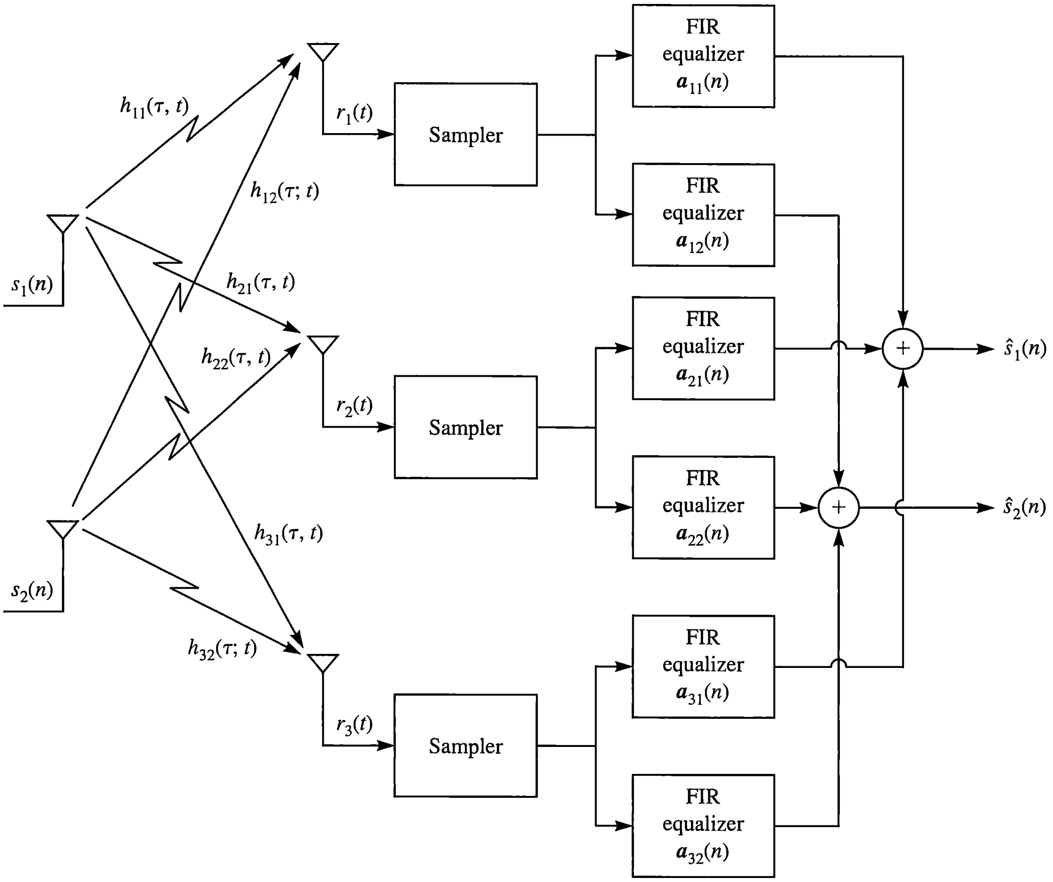

$h_{11}(\tau, t)$
$h_{12}(\tau; t)$
$h_{21}(\tau, t)$
$h_{22}(\tau, t)$
$h_{31}(\tau, t)$
$h_{32}(\tau; t)$
$s_1(n)$
$s_2(n)$
$r_1(t)$
$r_2(t)$
$r_3(t)$
Sampler
Sampler
Sampler
FIR equalizer $a_{11}(n)$
FIR equalizer $a_{12}(n)$
FIR equalizer $a_{21}(n)$
FIR equalizer $a_{22}(n)$
FIR equalizer $a_{31}(n)$
FIR equalizer $a_{32}(n)$
+
+
$\hat{s}_1(n)$
$\hat{s}_2(n)$

FIGURE 15.1–6
Signal demodulation with linear equalizers for the frequency-selective channel.

and the equalizer coefficients are adjusted to minimize

$$\mathrm{MSE}_j = E\left[|e_j(n)|^2\right], \qquad j = 1, 2, \ldots, N_T \qquad (15.1\text{–}28)$$

Either the LMS algorithm or the RLS algorithm described in Sections 10.1 and 10.4 may be used to adjust the equalizer coefficients. Following the training symbols, in the data transmission mode, the detector outputs may be used in place of the training symbols to form the error signal, i.e.,

$$e_j(n) = \tilde{s}_j(n) - \hat{s}_j(n), \qquad j = 1, 2, \ldots, N_T \qquad (15.1\text{–}29)$$

where $\tilde{s}_j(n)$ is the output of the detector for the jth symbol at time n, which is the symbol nearest in distance to the estimate $\hat{s}_j(n)$.

EXAMPLE 15.1–1. Consider a MIMO system in which the channel impulse responses are

$$h_{ij}(\tau; t) = h_{ij}^{(1)}\delta(\tau) + h_{ij}^{(2)}\delta(\tau - T), \qquad \begin{array}{l} i = 1, 2, \ldots, N_R \\ j = 1, 2, \ldots, N_T \end{array}$$

where T is the symbol interval. In this case, the channel is time dispersive with intersymbol interference occurring over two successive symbols. The channel coefficients

$\{h_{ij}^{(1)}\}$ and $\{h_{ij}^{(2)}\}$ are assumed to be fixed over a time interval spanning 2000 symbols, and are zero-mean complex-valued Gaussian random variables with variances

$$\sigma_{ij}^2(k) = E\left[\left|h_{ij}^{(k)}\right|^2\right], \qquad k = 1, 2$$

The sum of all these variances is normalized to unity, i.e.,

$$\sum_{k=1}^{2}\sum_{j=1}^{N_T}\sum_{i=1}^{N_R}\sigma_{ij}^2(k) = 1$$

A Monte Carlo simulation of the performance of the linear equalizers for the case in which the two multipath components have equal variance and the modulation is binary PSK is shown in Figure 15.1–7 for $(N_T, N_R) = (1, 1)$, $(2, 2)$, and $(2, 3)$. The linear equalizers were trained initially with the LMS algorithm for 1000 symbols. The simulations were performed for 1000 different channel realizations. The maximum achievable diversity is $2N_R$, where the factor of 2 is due to the multipath.

We observe that the effect of the ISI in the performance of the MIMO system is very severe. There is a significant loss in the performance of the (2, 2) and (2, 3) MIMO

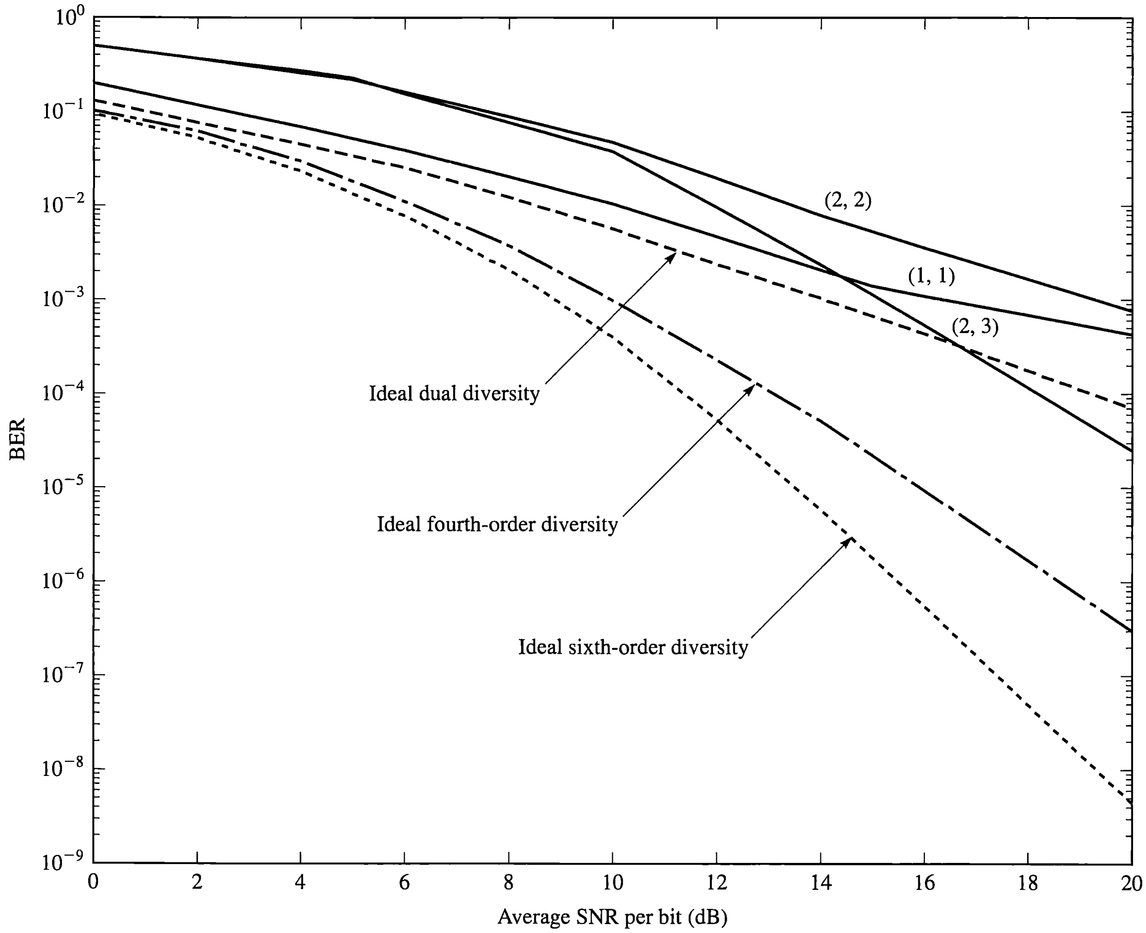


FIGURE 15.1–7

Performance of linear equalizer for two-path channel with (N_T, N_R) antennas for spatial multiplexing.

systems due to the ISI. This effect is due to the basic limitation of linear equalizers to mitigate ISI in fading multipath channels.

Other Equalizer Structures

The linear adaptive equalizer described above for the MIMO channel is the simplest equalization technique from the viewpoint of computational complexity. To achieve better performance, one may employ a more powerful equalizer, in particular, a decision-feedback equalizer (DFE) or a maximum-likelihood sequence detector (MLSD).

Figure 15.1–8 illustrates the structure of a DFE for a MIMO channel with $N_T = N_R = 2$ antennas. The two feedforward filters at each receive antenna are structurally identical to the FIR filters in a linear equalizer structure. Typically, these FIR filters have fractionally spaced taps. The two feedback filters connected to each detector are symbol-spaced FIR filters. Their function is to suppress the ISI that is inherent in previously detected symbols (so-called postcursors). Thus, the estimate of the jth information symbol transmitted at time instant n may be expressed as

$$\hat{s}_j(n) = \sum_{i=1}^{N_R} \left\{ \sum_{k=-K_1}^{0} a_{ij}(k; n) r_i(n-k) - \sum_{k=1}^{K_2} b_{ij}(k; n) \tilde{s}_i(n-k) \right\} \qquad (15.1\text{–}30)$$

where $K_1 + 1$ is the number of tap coefficients in each of the feedforward filters and K_2 is the number of tap coefficients $\{b_{ij}(k; n)\}$ in each of the feedback filters.

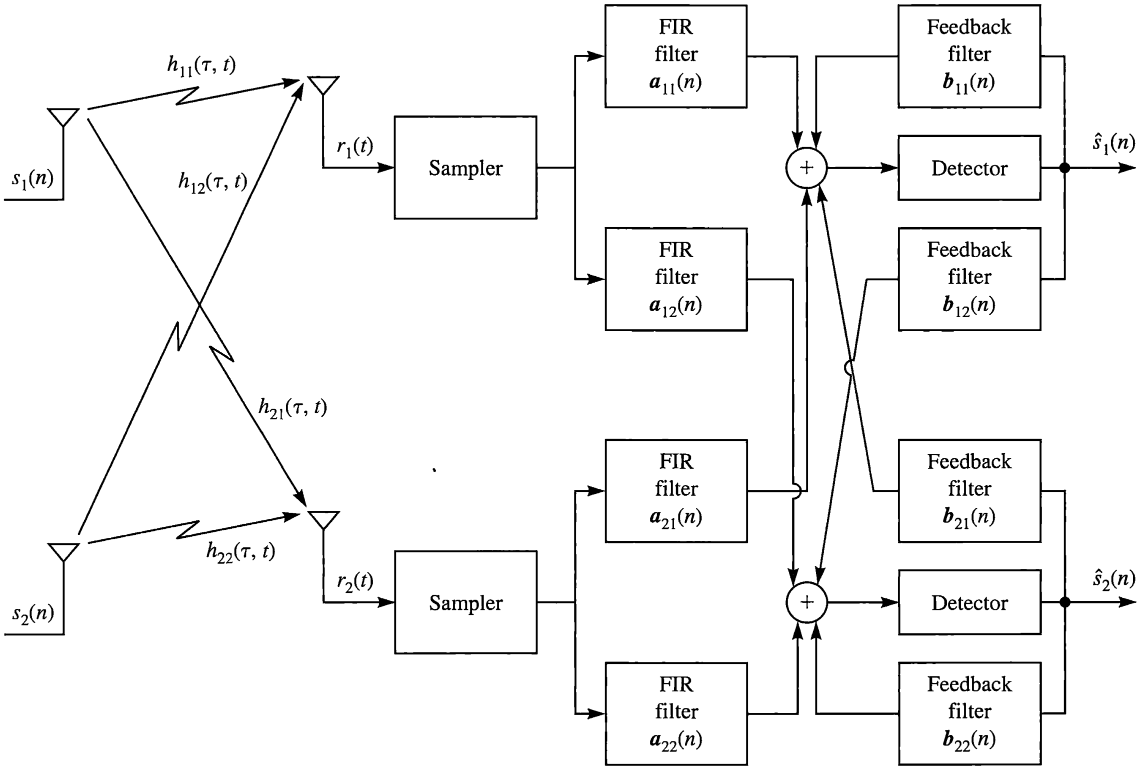

$h_{11}(\tau, t)$
$h_{12}(\tau, t)$
$h_{21}(\tau, t)$
$h_{22}(\tau, t)$
$s_1(n)$
$s_2(n)$
$r_1(t)$
$r_2(t)$
Sampler
Sampler
FIR filter $a_{11}(n)$
FIR filter $a_{12}(n)$
FIR filter $a_{21}(n)$
FIR filter $a_{22}(n)$
Feedback filter $b_{11}(n)$
Feedback filter $b_{12}(n)$
Feedback filter $b_{21}(n)$
Feedback filter $b_{22}(n)$
+
+
Detector
Detector
$\hat{s}_1(n)$
$\hat{s}_2(n)$

FIGURE 15.1–8

Signal demodulation with decision-feedback equalizers for the frequency-selective channel.

As in the case of the linear equalizers for the MIMO channel, the MSE criterion may be used to adjust the coefficients of the feedforward and feedback filters. Training symbols are usually needed to adjust the equalizer coefficients initially. When data are transmitted in frames, training symbols may be inserted in each frame for initial adjustment of the DFE coefficients. During the transmission of information symbols, the symbols at the output of the detector may be used for coefficient adjustment. We note that the computational complexity of the DFE is comparable to that of the linear MIMO equalizer.

EXAMPLE 15.1-2. Consider the MIMO system described in Example 15.1–1, where the linear equalizers are replaced by decision-feedback equalizers. The error rate performance of the MIMO system with DFEs, obtained by Monte Carlo simulation, is shown in Figure 15.1–9. In comparing the performance of the MIMO system with DFEs and with linear equalizers, we observe that the DFEs generally yield better performance. Nevertheless, there is still a significant loss in performance due to ISI.

The best performance in the presence of ISI is obtained when the equalization algorithm is based on the MLSD criterion. A multichannel version of the Viterbi algorithm

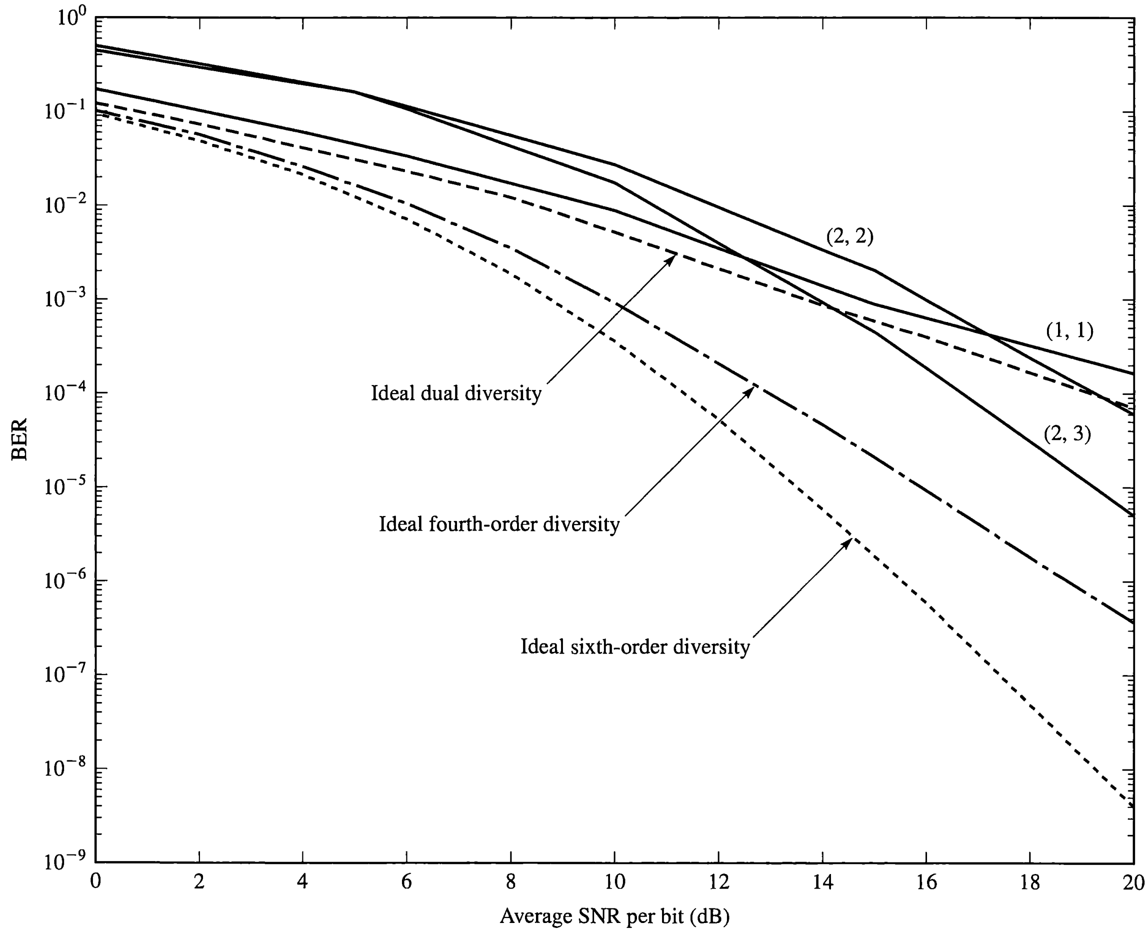


FIGURE 15.1–9
Performance of DFEs for two-path channel with (N_T, N_R) antennas for spatial multiplexing.

is computationally efficient in implementing MLSD for a MIMO channel with ISI. The major impediment in the implementation of the Viterbi algorithm is its computational complexity, which grows exponentially as M^L, where M is the size of the symbol constellation and L is the span of the channel multipath dispersion expressed in terms of the number of information symbols spanned. Consequently, except for channels with relatively small multipath spread, e.g., $L = 2$ or 3, and small signal constellations, e.g., $M = 2$ or 4, the implementation complexity of the Viterbi algorithm for a MIMO system is very high compared to that for a DFE.

15.2

CAPACITY OF MIMO CHANNELS

In this section, we evaluate the capacity of MIMO channel models. For mathematical convenience, we limit our treatment to frequency-nonselective channels which are assumed to be known to the receiver. Thus, the channel is characterized by an $N_R \times N_T$ channel matrix $\boldsymbol{H}$ with elements $\{h_{ij}\}$. In any signal interval, the elements $\{h_{ij}\}$ are complex-valued random variables. In the special case of a Rayleigh fading channel, the $\{h_{ij}\}$ are zero-mean complex-valued Gaussian random variables with uncorrelated real and imaginary components (circularly symmetric). When the $\{h_{ij}\}$ are statistically independent and identically distributed complex-valued Gaussian random variables, the MIMO channel is spatially white.

15.2–1 Mathematical Preliminaries

By using a singular value decomposition (SVD), the channel matrix $\boldsymbol{H}$ with rank r may be expressed as

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H \tag{15.2–1}$$

where $\boldsymbol{U}$ is an $N_R \times r$ matrix, $\boldsymbol{V}$ is an $N_T \times r$ matrix, and $\boldsymbol{\Sigma}$ is an $r \times r$ diagonal matrix with diagonal elements the singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$ of the channel. The singular values $\{\sigma_i\}$ are strictly positive and are ordered in decreasing order, i.e., $\sigma_i \geq \sigma_{i+1}$. The column vectors of $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthonormal. Hence $\boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{I}_r$ and $\boldsymbol{V}^H\boldsymbol{V} = \boldsymbol{I}_r$, where $\boldsymbol{I}_r$ is an $r \times r$ identity matrix. Therefore, the SVD of the channel matrix $\boldsymbol{H}$ may be expressed as

$$\boldsymbol{H} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^H \tag{15.2–2}$$

where $\{\boldsymbol{u}_i\}$ are the column vectors of $\boldsymbol{U}$, which are called the *left singular vectors* of $\boldsymbol{H}$, and $\{\boldsymbol{v}_i\}$ are the column vectors of $\boldsymbol{V}$, which are called the *right singular vectors* of $\boldsymbol{H}$.

We also consider the decomposition of the $N_R \times N_R$ square matrix $\boldsymbol{H}\boldsymbol{H}^H$. This matrix may be decomposed as

$$\boldsymbol{H}\boldsymbol{H}^H = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^H \tag{15.2–3}$$

where Q is the $N_R \times N_R$ modal matrix with orthonormal column vectors (eigenvectors), i.e., $Q^H Q = I_{N_R}$, and Λ is an $N_R \times N_R$ diagonal matrix with diagonal elements $\{\lambda_i, i = 1, 2, \ldots, N_R\}$, which are the eigenvalues of HH^H. With the eigenvalues numbered in decreasing order ($\lambda_i \geq \lambda_{i+1}$), it can be easily demonstrated that the eigenvalues of HH^H are related to the singular values in the SVD of H as follows:

$$\lambda_i = \begin{cases} \sigma_i^2 & i = 1, 2, \ldots, r \\ 0 & i = r+1, \ldots, N_R \end{cases} \tag{15.2–4}$$

A useful metric is the Frobenius norm of H, which is defined as

$$\begin{aligned} \| H \|_F &= \sqrt{\sum_{i=1}^{N_R} \sum_{j=1}^{N_T} |h_{ij}|^2} \\ &= \sqrt{\text{trace}\,(HH^H)} \qquad\qquad (15.2\text{–}5) \\ &= \sqrt{\sum_{i=1}^{N_R} \lambda_i} \end{aligned}$$

We shall observe below that the squared Frobenius norm $\|H\|_F^2$ is a parameter that determines the performance of MIMO communication systems. The statistical properties of $\|H\|_F^2$ can be determined for various fading channel conditions. For example, in the case of Rayleigh fading, $|h_{ij}|^2$ is a chi-squared random variable with two degrees of freedom. When the $\{h_{ij}\}$ are iid (spatially white MIMO channel) with unit variance, the probability density function of $\|H\|_F^2$ is chi-squared with $2N_R N_T$ degrees of freedom; i.e., if $X = \|H\|_F^2$,

$$p(x) = \frac{x^{n-1}}{(n-1)!} e^{-x}, \qquad x \geq 0 \tag{15.2–6}$$

where $n = N_R N_T$.

15.2–2 Capacity of a Frequency-Nonselective Deterministic MIMO Channel

Let us consider a frequency-nonselective AWGN MIMO channel characterized by the matrix H. Let s denote the $N_T \times 1$ transmitted signal vector, which is statistically stationary and has zero mean and autocovariance matrix R_{ss}. In the presence of AWGN, the $N_R \times 1$ received signal vector y may be expressed as

$$y = Hs + \eta \tag{15.2–7}$$

where η is the $N_R \times 1$ zero-mean Gaussian noise vector with covariance matrix $R_{nn} = N_0 I_{N_R}$. Although H is a realization of a random matrix, in this section we treat H as deterministic and known to the receiver.

To determine the capacity of the MIMO channel, we first compute the mutual information between the transmitted signal vector $\boldsymbol{s}$ and the received vector $\boldsymbol{y}$, denoted as $I(\boldsymbol{s}; \boldsymbol{y})$, and then determine the probability distribution of the signal vector $\boldsymbol{s}$ that maximizes $I(\boldsymbol{s}; \boldsymbol{y})$. Thus,

$$C = \max_{p(\boldsymbol{s})} I(\boldsymbol{s}; \boldsymbol{y}) \tag{15.2–8}$$

where C is the channel capacity in bits per second per hertz (bps/Hz). It can be shown (see Telatar (1999) and Neeser and Massey (1993)) that $I(\boldsymbol{s}; \boldsymbol{y})$ is maximized when $\boldsymbol{s}$ is a zero-mean, circularly symmetric, complex Gaussian vector; hence, C is only dependent on the covariance of the signal vector. The resulting capacity of the MIMO channel is

$$C = \max_{\operatorname{tr}(\boldsymbol{R}_{ss})=\mathcal{E}_s} \log_2 \det\left(\boldsymbol{I}_{N_R} + \frac{1}{N_0}\boldsymbol{H}\boldsymbol{R}_{ss}\boldsymbol{H}^H\right) \quad \text{bps/Hz} \tag{15.2–9}$$

where $\operatorname{tr}(\boldsymbol{R}_{ss})$ denotes the trace of the signal covariance $\boldsymbol{R}_{ss}$. This is the maximum rate per hertz that can be transmitted reliably (without errors) over the MIMO channel for any given realization of the channel matrix $\boldsymbol{H}$.

In the important practical case where the signals among the N_T transmitters are statistically independent symbols with energy per symbol equal to $\mathcal{E}_s/N_T$, the signal covariance matrix is diagonal, i.e.,

$$\boldsymbol{R}_{ss} = \frac{\mathcal{E}_s}{N_T}\boldsymbol{I}_{N_T} \tag{15.2–10}$$

and trace $(\boldsymbol{R}_{ss}) = \mathcal{E}_s$. In this case, the expression for the capacity of the MIMO channel simplifies to

$$C = \log_2 \det\left(\boldsymbol{I}_{N_R} + \frac{\mathcal{E}_s}{N_T N_0}\boldsymbol{H}\boldsymbol{H}^H\right) \quad \text{bps/Hz} \tag{15.2–11}$$

The capacity formula in Equation 15.2–11 can also be expressed in terms of the eigenvalues of $\boldsymbol{H}\boldsymbol{H}^H$ by using the decomposition $\boldsymbol{H}\boldsymbol{H}^H = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^H$. Thus,

$$\begin{aligned} C &= \log_2 \det\left(\boldsymbol{I}_{N_R} + \frac{\mathcal{E}_s}{N_T N_0}\boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^H\right) \\ &= \log_2 \det\left(\boldsymbol{I}_{N_R} + \frac{\mathcal{E}_s}{N_T N_0}\boldsymbol{Q}^H\boldsymbol{Q}\boldsymbol{\Lambda}\right) \\ &= \log_2 \det\left(\boldsymbol{I}_{N_R} + \frac{\mathcal{E}_s}{N_T N_0}\boldsymbol{\Lambda}\right) \\ &= \sum_{i=1}^{r} \log_2\left(1 + \frac{\mathcal{E}_s}{N_T N_0}\lambda_i\right) \end{aligned} \tag{15.2–12}$$

where r is the rank of the channel matrix $\boldsymbol{H}$.

It is interesting to note that in a SISO channel, $\lambda_1 = |h_{11}|^2$ so that

$$C_{\text{SISO}} = \log_2\left(1 + \frac{\mathcal{E}_s}{N_0}|h_{11}|^2\right) \qquad \text{bps/Hz} \tag{15.2–13}$$

We observe that the capacity of the MIMO channel is simply equal to the sum of the capacities of r SISO channels, where the transmit energy per SISO channel is $\mathcal{E}_s/N_T$ and the corresponding channel gain is equal to the eigenvalue λ_i.

Capacity of SIMO Channel

A SIMO channel ($N_T = 1,\ N_R \geq 2$) is characterized by the vector $\boldsymbol{h} = [h_{11}\, h_{21} \ldots h_{N_R 1}]^t$. In this case, the rank of the channel matrix is unity, and the eigenvalue λ_1 is given as

$$\lambda_1 = \| \boldsymbol{h} \|_F^2 = \sum_{i=1}^{N_R} |h_{i1}|^2 \tag{15.2–14}$$

Therefore, the capacity of the SIMO channel, when the N_R elements $\{h_{i1}\}$ of the channel are deterministic and known to the receiver, is

$$\begin{aligned} C_{\text{SIMO}} &= \log_2\left(1 + \frac{\mathcal{E}_s}{N_0} \| \boldsymbol{h} \|_F^2\right) \\ &= \log_2\left(1 + \frac{\mathcal{E}_s}{N_0} \sum_{i=1}^{N_R} |h_{i1}|^2\right) \qquad \text{bps/Hz} \end{aligned} \tag{15.2–15}$$

Capacity of MISO Channel

A MISO channel ($N_T \geq 2$, $N_R = 1$) is characterized by the vector $\boldsymbol{h} = [h_{11}\, h_{12} \ldots h_{1N_T}]^t$. In this case, the rank of the channel matrix is also unity, and the eigenvalue λ_1 is given as

$$\lambda_1 = \| \boldsymbol{h} \|_F^2 = \sum_{j=1}^{N_T} |h_{1j}|^2 \tag{15.2–16}$$

The resulting capacity of the MISO channel when the N_T elements $\{h_{1j}\}$ of the channel are deterministic and known to the receiver is

$$\begin{aligned} C_{\text{MISO}} &= \log_2\left(1 + \frac{\mathcal{E}_s}{N_T N_0} \| \boldsymbol{h} \|_F^2\right) \\ &= \log_2\left(1 + \frac{\mathcal{E}_s}{N_T N_0} \sum_{j=1}^{N_T} |h_{1j}|^2\right) \qquad \text{bps/Hz} \end{aligned} \tag{15.2–17}$$

It is interesting to note that for the same $\| \boldsymbol{h} \|_F^2$, the capacity of the SIMO channel is greater than the capacity of the MISO channel when the channel is known to the receiver only. The reason is that, under the constraint that the total transmitted energy in the two systems be identical, the energy $\mathcal{E}_s$ in the MISO system is split evenly among the N_T transmit antennas, whereas in the SIMO system, the transmitter energy $\mathcal{E}_s$ is used by the single antenna. Note also that in both SIMO and MISO channels, the capacity grows logarithmically as a function of $\| \boldsymbol{h} \|_F^2$.

15.2–3 Capacity of a Frequency-Nonselective Ergodic Random MIMO Channel

The channel capacity expressions derived in Section 15.2–2 for a deterministic MIMO channel may be viewed as the capacity for a randomly selected realization of the channel matrix. To determine the ergodic capacity, we may simply average the expression for the capacity of the deterministic channel over the statistics of the channel matrix. Thus, for a SIMO channel, the ergodic capacity, as defined in Chapter 14, is

$$\begin{aligned}\bar{C}_{\text{SIMO}} &= E\left[\log_2\left(1+\frac{\mathcal{E}_s}{N_0}\sum_{i=1}^{N_R}|h_{i1}|^2\right)\right] \\ &= \int_0^\infty \log_2\left(1+\frac{\mathcal{E}_s}{N_0}x\right)p(x)\,dx \qquad \text{bps/Hz}\end{aligned} \tag{15.2–18}$$

where $X = \Sigma_{i=1}^{N_R}|h_{i1}|^2$ and $p(x)$ is the probability density function of the random variable X.

Figure 15.2–1 illustrates $\bar{C}_{\text{SIMO}}$ versus the average SNR $\mathcal{E}_s E(|h_{i1}|^2)/N_0$ for $N_R = 2, 4$, and 8 when the channel parameters $\{h_{i1}\}$ are iid complex-valued, zero-mean, circularly symmetric Gaussian with each having unit variance. Hence, the random

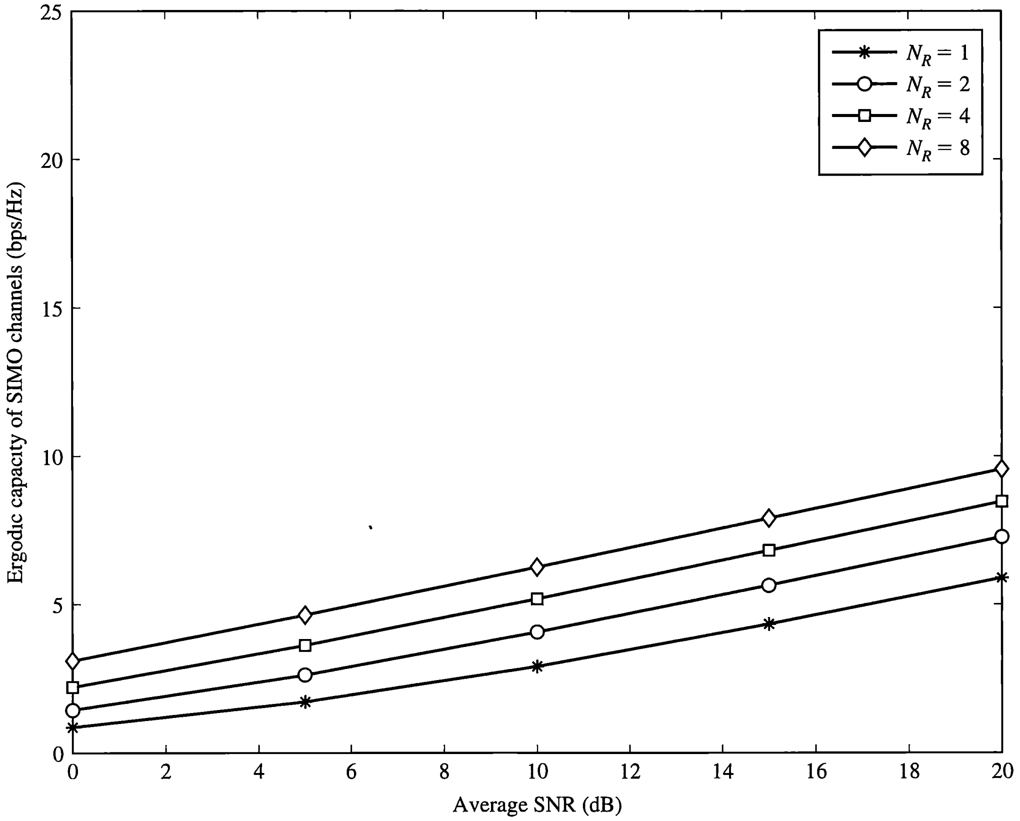


FIGURE 15.2–1
Ergodic capacity of SIMO channels.

variable X has a chi-squared distribution with $2N_R$ degrees of freedom, and its PDF is given by Equation 15.2–6. For comparison, the ergodic capacity $\bar{C}_{\text{SISO}}$ is also shown.

Similarly, the ergodic channel capacity for the MISO channel is

$$\begin{aligned}\bar{C}_{\text{MISO}} &= E\left[\log_2\left(1+\frac{\mathcal{E}_s}{N_T N_0}\sum_{j=1}^{N_T}|h_{1j}|^2\right)\right] \\ &= \int_0^\infty \log_2\left(1+\frac{\mathcal{E}_s}{N_T N_0}x\right)p(x)\,dx \qquad \text{bps/Hz}\end{aligned} \tag{15.2–19}$$

Figure 15.2–2 illustrates $\bar{C}_{\text{MISO}}$ versus the average SNR, as defined above, for $N_T = 2, 4$, and 8 when the channel parameters $\{h_{1j}\}$ are iid zero-mean, complex-valued, circularly symmetric Gaussian, each having unit variance. As in the case of the SIMO channel, the random variable x has a chi-squared distribution with $2N_T$ degrees of freedom. The ergodic capacity of a SISO channel is also included in Figure 15.2–2 for comparison purposes. In comparing the graphs in Figure 15.2–1 with those in Figure 15.2–2, we observe that $\bar{C}_{\text{SIMO}} > \bar{C}_{\text{MISO}}$.

To determine the ergodic capacity of the MIMO channel, we average the expression for C given in Equation 15.2–12 over the joint probability density function of the

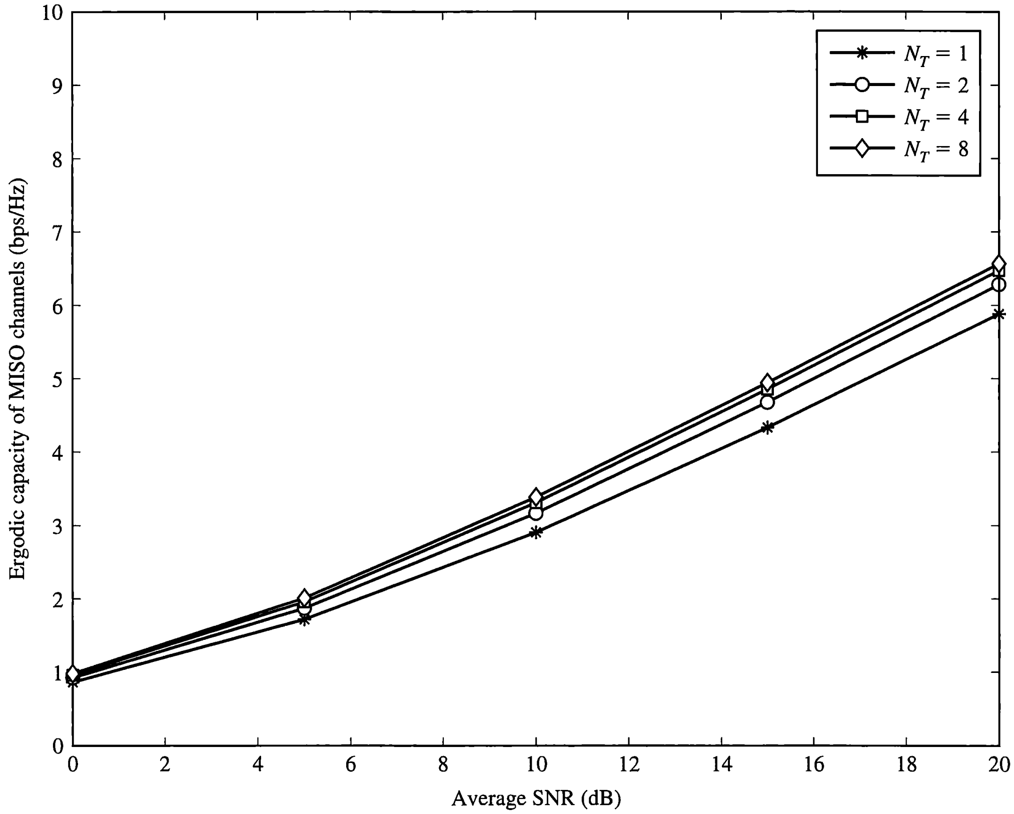


FIGURE 15.2–2
Ergodic capacity of MISO channels.

eigenvalues $\{\lambda_i\}$. Thus,

$$\begin{aligned}\bar{C}_{\text{MIMO}} &= E\left\{\sum_{i=1}^{r} \log_2\left(1+\frac{\mathcal{E}_s}{N_T N_0}\lambda_i\right)\right\} \\ &= \int_0^{\infty}\cdots\int_0^{\infty}\left[\sum_{i=1}^{r}\log_2\left(1+\frac{\mathcal{E}_s}{N_T N_0}\lambda_i\right)\right] p(\lambda_1,\ldots,\lambda_r)\,d\lambda_1\cdots d\lambda_r\end{aligned} \tag{15.2–20}$$

For the case in which the elements of the channel matrix $\boldsymbol{H}$ are complex-valued zero-mean Gaussian with unit variance and spatially white with $N_R = N_T = N$, the joint PDF of $\{\lambda_i\}$ is given by Edelman (1989) as

$$p(\lambda_1, \lambda_2, \ldots, \lambda_N) = \frac{(\pi/2)^{N(N-1)}}{[\Gamma_N(N)]^2}\exp\left[-\left(\sum_{i=1}^{N}\lambda_i\right)\right]\prod_{\substack{i,j\\ i<j}}(2\lambda_i - 2\lambda_j)^2\prod_{i=1}^{N}u(\lambda_i) \tag{15.2–21}$$

where $\Gamma_N(N)$ is the multivariate gamma function defined as

$$\Gamma_N(N) = \pi^{N(N-1)/2}\prod_{i=1}^{N}(N-i)! \tag{15.2–22}$$

Figure 15.2–3 illustrates $\bar{C}_{\text{MIMO}}$ versus the average SNR for $N_T = N_R = 2$ and $N_T = N_R = 4$. The ergodic capacity of a SISO channel is also included in Figure 15.2–3 for comparison purposes. We observe that at high SNRs, the capacity of the $(N_T, N_R) = (4, 4)$ MIMO system is approximately four times the capacity of the $(1, 1)$ system. Thus, at high SNRs, the capacity increases linearly with the number of antenna pairs when the channel is spatially white.

15.2–4 Outage Capacity

As we have observed, the capacity of a randomly fading channel is a random variable. For an ergodic channel, its average value $\bar{C}$ is the ergodic capacity. For a nonergodic channel, a useful performance metric is the probability that the capacity is below some value for a specified percentage of channel realizations. This performance metric is the outage capacity, defined in Section 14.2–2.

To be specific, we consider a channel that is known to the receiver only. We assume that the MIMO channel matrix $\boldsymbol{H}$ is randomly selected in accordance with each channel realization and remains constant for each channel use. In other words, we assume that the channel is quasi-static for the duration of a frame of data, but the channel matrix may change from frame to frame. Then, for any given frame, the probability

$$P(C \leq C_p) = P_{\text{out}} \tag{15.2–23}$$

is called the *outage probability* and the corresponding capacity C_p is called the 100 P_{out}% *outage capacity* where the subscript p denotes P_{out}. Hence, the achievable information

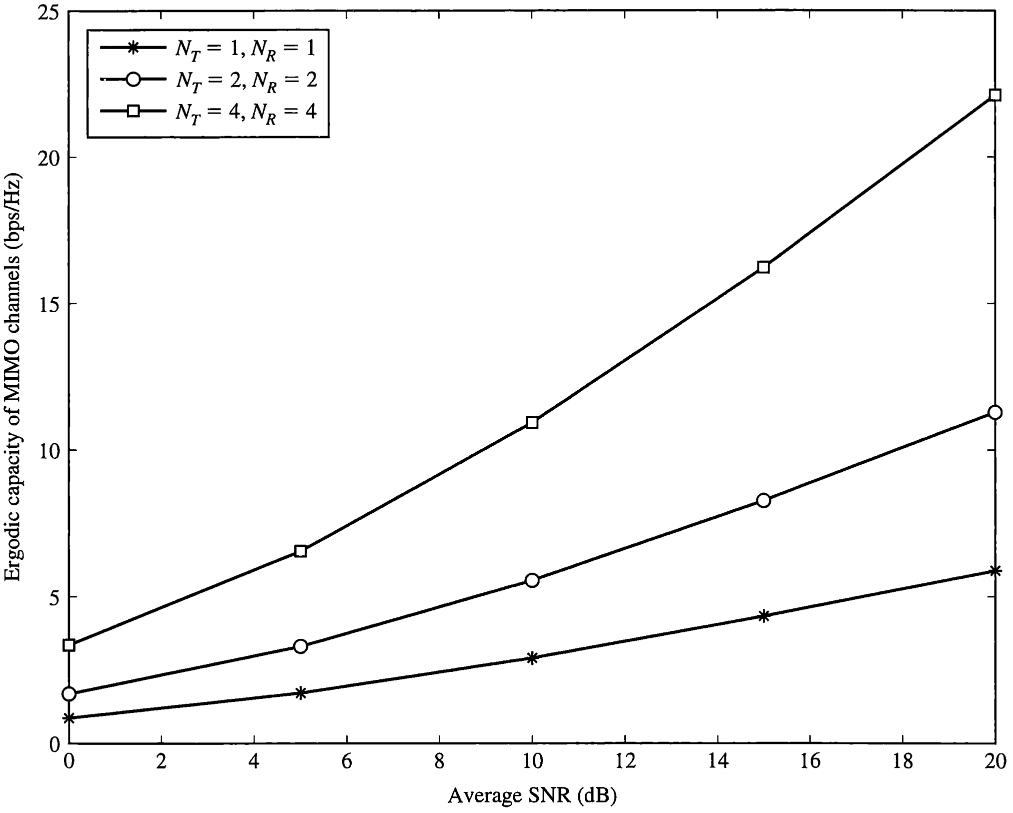


FIGURE 15.2–3
Ergodic capacity of MIMO channels.

rate will exceed C_p for $100(1 - P_{\text{out}})\%$ of the MIMO channel realizations. Equivalently, if we transmit a large number of frames, the transmission of a frame will fail (contain errors) with probability P_{out}.

To evaluate the outage capacity of a MIMO channel, let us consider a channel matrix $\boldsymbol{H}$, whose elements are iid, complex-valued, circularly symmetric, zero-mean Gaussian with unit variance. Then, for each realization of $\boldsymbol{H}$, say $\boldsymbol{H}_k$, the corresponding capacity C_k is given by Equation 15.2–11 for any SNR $\mathcal{E}_s/N_0$. If we consider the ensemble of all possible channel realizations for any given SNR, the PDF of C_k may appear as shown in Figure 15.2–4.

The cumulative distribution function (CDF) is

$$F(C) = P(C_k \leq C)$$

Figure 15.2–5 illustrates the CDF for $N_T = N_R = 2$ and $N_T = N_R = 4$ MIMO channels and a SISO channel for an SNR of 10 dB. The outage capacity at some specified outage probability is easily determined from $F(C)$ for any given SNR.

Figure 15.2–6 illustrates the 10% outage capacity as a function of the SNR for $N_T = N_R = 2$ and $N_T = N_R = 4$ MIMO channels and for a SISO channel. We observe that, as in the case of the ergodic capacity, the outage capacity increases as the SNR is increased and as the number of antennas $N_R = N_T$ increases.

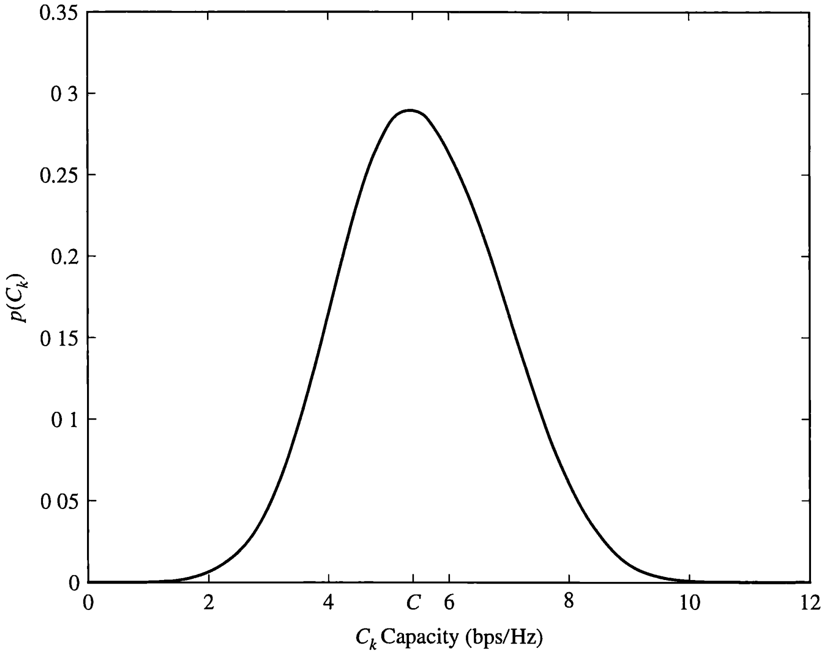


FIGURE 15.2–4
Probability density function of channel capacity for an $N_T = N_R = 2$ MIMO channel at SNR = 10 dB.

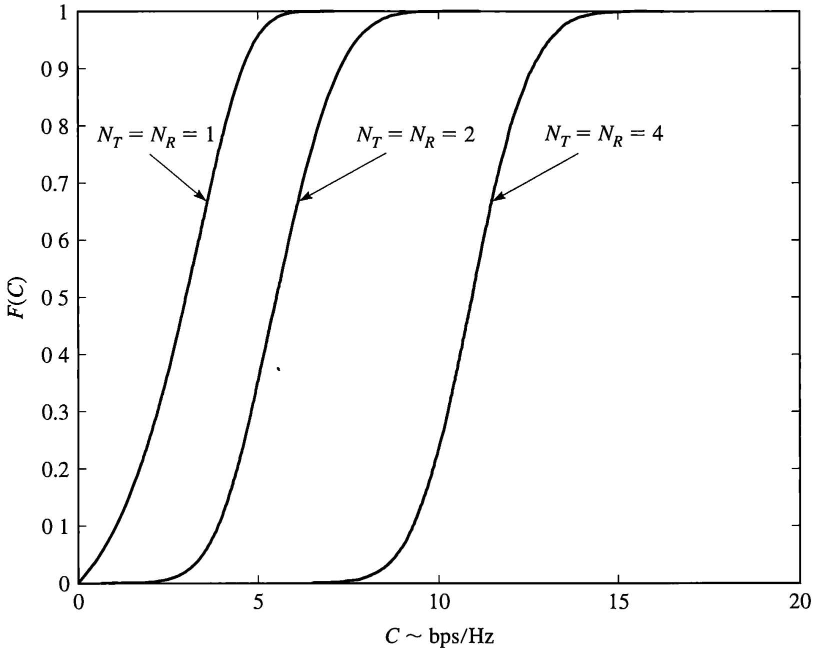


FIGURE 15.2–5
CDF of MIMO channel capacity at SNR = 10 dB.

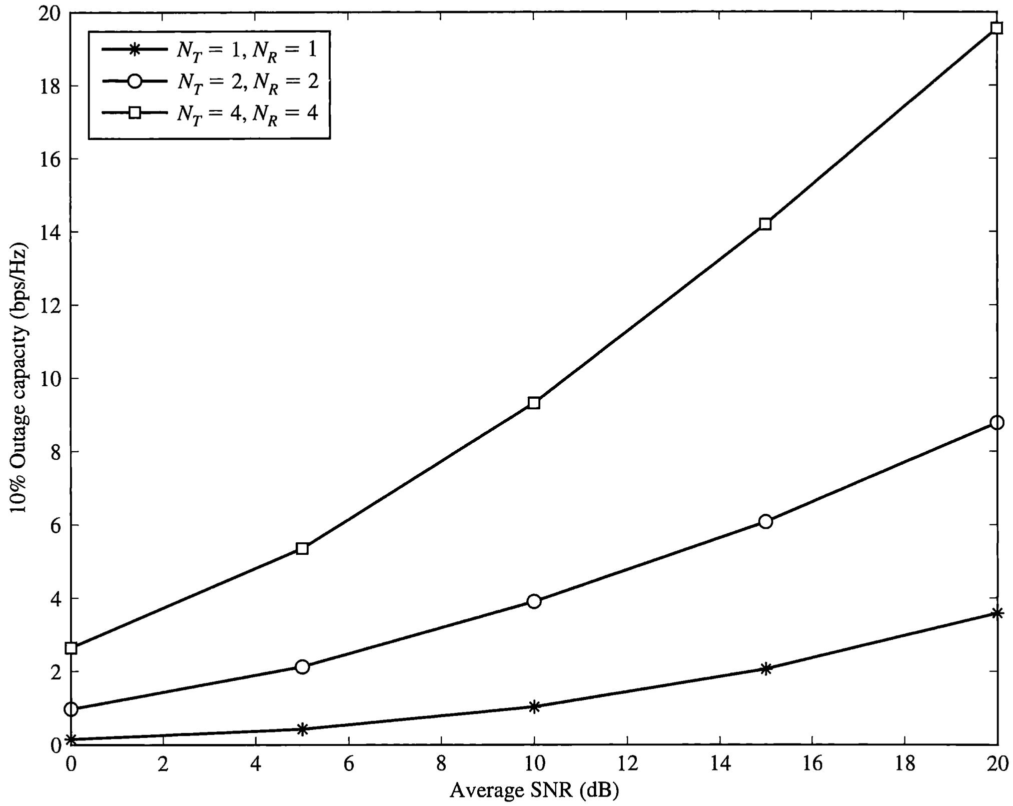


FIGURE 15.2–6
10% Outage capacity of MIMO channels.

15.2–5 Capacity of MIMO Channel When the Channel Is Known at the Transmitter

We have observed that when the channel matrix $\boldsymbol{H}$ is known only at the receiver, the transmitter allocates equal power to the signals transmitted on the multiple transmit antennas. On the other hand, if both the transmitter and the receiver know the channel matrix, the transmitter can allocate its transmitted power more efficiently and thus achieve a higher capacity.

Let us consider a MIMO system with N_T transmit antennas and N_R receive antennas in a frequency-nonselective channel. The channel matrix $\boldsymbol{H}$ is assumed to be of rank r. Hence, using an SVD, $\boldsymbol{H}$ is represented as $\boldsymbol{H} = \boldsymbol{U\Sigma V}^H$. Since $\boldsymbol{H}$ is known at the transmitter and the receiver, the transmitted signal vector of dimension $r \times 1$ is premultiplied by the matrix $\boldsymbol{V}$, and the received signal is premultiplied by the matrix $\boldsymbol{U}^H$ as previously described in Section 15.1–2 and in Figure 15.1–5. The transmitted signal vector $\boldsymbol{s}$ has zero-mean, complex-valued Gaussian elements. The sum of the variances of the elements of $\boldsymbol{s}$ is constrained to be equal to N_T, i.e.,

$$E(\boldsymbol{s}^H\boldsymbol{s}) = \sum_{k=1}^{r} E\left[|s_k|^2\right] = \sum_{k=1}^{r} \sigma_{ks}^2 = N_T \tag{15.2–24}$$

Hence, the signal transmitted on the N_T antennas is $\sqrt{\mathcal{E}_s/N_T}\,\boldsymbol{V s}$.

The received signal vector is

$$y = \sqrt{\frac{\mathcal{E}_s}{N_T}} \boldsymbol{H}\boldsymbol{V}\boldsymbol{s} + \boldsymbol{\eta} \boldsymbol{y} = \sqrt{\frac{\mathcal{E}_s}{N_T}} \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{s} + \boldsymbol{\eta} \tag{15.2–25}$$

After premultiplying $\boldsymbol{y}$ by $\boldsymbol{U}^H$, we obtain the transformed $r \times 1$ vector

$$\boldsymbol{y}' = \boldsymbol{U}^H \boldsymbol{y} = \sqrt{\frac{\mathcal{E}_s}{N_T}} \boldsymbol{\Sigma}\boldsymbol{s} + \boldsymbol{\eta}' \tag{15.2–26}$$

where $\boldsymbol{\eta}' = \boldsymbol{U}^H \boldsymbol{\eta}$.

We observe that the channel characterized by the $N_R \times N_T$ channel matrix is equivalent to r decoupled SISO channels, whose output is

$$y_k' = \sqrt{\frac{\mathcal{E}_s \lambda_k}{N_T}} s_k + \eta_k', \qquad k = 1, 2, \ldots, r \tag{15.2–27}$$

Therefore, the capacity of the MIMO channel for a specific power allocation at the transmitter is

$$C\left(\{\sigma_{ks}^2\}\right) = \sum_{k=1}^{r} \log_2\left(1 + \frac{\mathcal{E}_s \lambda_k}{N_T N_0} \sigma_{ks}^2\right) \tag{15.2–28}$$

Note that the energy transmitted per symbol on the kth subchannel is $\mathcal{E}_s \sigma_{ks}^2 / N_T$. The transmitter allocates its total transmitted power across the N_T antennas so as to maximize $C\left(\{\sigma_{ks}^2\}\right)$. Thus, the capacity of the MIMO channel under the optimum power allocation is

$$C = \max_{\{\sigma_{ks}^2\}} \sum_{k=1}^{r} \log_2\left(1 + \frac{\mathcal{E}_s \lambda_k}{N_T N_0} \sigma_{ks}^2\right) \tag{15.2–29}$$

where the constraint on the$\{\sigma_{ks}^2\}$ is given by Equation 15.2–24. The maximization in Equation 15.2–29 can be performed by numerical methods. Basically, the solution satisfies the "water-filling principle," which allocates more power to subchannels which have low noise power, i.e., according to the ratio N_0/λ_k, and less power to subchannels that have high noise power.

For an ergodic channel, the average (ergodic) capacity, is determined by averaging the capacity given in Equation 15.2–29 for a given $\boldsymbol{H}$ over the channel statistics, i.e., over the joint PDF of $\{\lambda_k\}$. Thus,

$$\bar{C} = E\left\{\max_{\{\sigma_{ks}^2\}} \sum_{k=1}^{r} \log_2\left(1 + \frac{\mathcal{E}_s \lambda_k}{N_T N_0} \sigma_{ks}^2\right)\right\} \tag{15.2–30}$$

This computation can be performed numerically when the joint PDF of the eigenvalues $\{\lambda_k\}$ is known.

15.3

SPREAD SPECTRUM SIGNALS AND MULTICODE TRANSMISSION

In Section 15.1 we demonstrated that a MIMO system transmitting in a frequency-nonselective fading channel can employ identical narrowband signals for data transmission. The signals from the N_T transmit antennas were assumed to arrive at the N_R receive antennas via $N_T N_R$ independently fading propagation paths. By knowing the channel matrix $\boldsymbol{H}$, the receiver is able to separate and detect the N_T transmitted symbols in each signaling interval. Thus, the use of narrowband signals provided a data rate increase (spatial multiplexing gain) of N_T relative to a single-antenna system and, simultaneously, a signal diversity of order N_R, where $N_R \geq N_T$, when the maximum-likelihood detector is employed.

In this section we consider a similar MIMO system with the exception that the transmitted signals on the N_T transmit antennas will be wideband, i.e., spread spectrum signals.

15.3–1 Orthogonal Spreading Sequences

The MIMO system under consideration is illustrated in Figure 15.3–1(a). The data symbols $\{s_j, 1 \leq j \leq N_T\}$ are each multiplied (spread) by a binary sequence $\{c_{jk}, 1 \leq k \leq L_c, 1 \leq j \leq N_T\}$ consisting of L_c bits, where each bit takes a value of either $+1$ or -1. These binary sequences are assumed to be orthogonal, i.e.,

$$\sum_{k=1}^{L_c} c_{jk} c_{ik} = 0, \qquad j \neq i \tag{15.3–1}$$

For example, the orthogonal sequences may be generated from N_T Hadamard codewords of block length L_c, where a 0 in the Hadamard codeword is mapped into a -1 and a 1 is mapped into a $+1$. The resulting orthogonal sequences are usually called Walsh-Hadamard sequences.

The transmitted signal on the jth transmit antenna may be expressed as

$$s_j(t) = s_j \sqrt{\frac{\mathcal{E}_s}{N_T}} \sum_{k=1}^{L_c} c_{jk} g(t - kT_c), \qquad 0 \leq t \leq T; \;\; j = 1, 2, \ldots, N_T \tag{15.3–2}$$

where $\mathcal{E}_s/N_T$ is the energy per transmitted symbol, T is the symbol duration, $T_c = T/L_c$, and $g(t)$ is a signal pulse of duration T_c and energy $1/L_c$. The pulse $g(t)$ is usually called a *chip*, and L_c is the number of chips per information symbol. Thus, the bandwidth of the information symbols, which is approximately $1/T$, is expanded by the factor L_c, so that the transmitted signal on each antenna occupies a bandwidth of approximately $1/T_c$.

The MIMO channel is assumed to be frequency-nonselective and characterized by the matrix $\boldsymbol{H}$, which is known to the receiver. At each receiving terminal, the received signal is passed through a chip matched filter and matched to the chip pulse $g(t)$, and

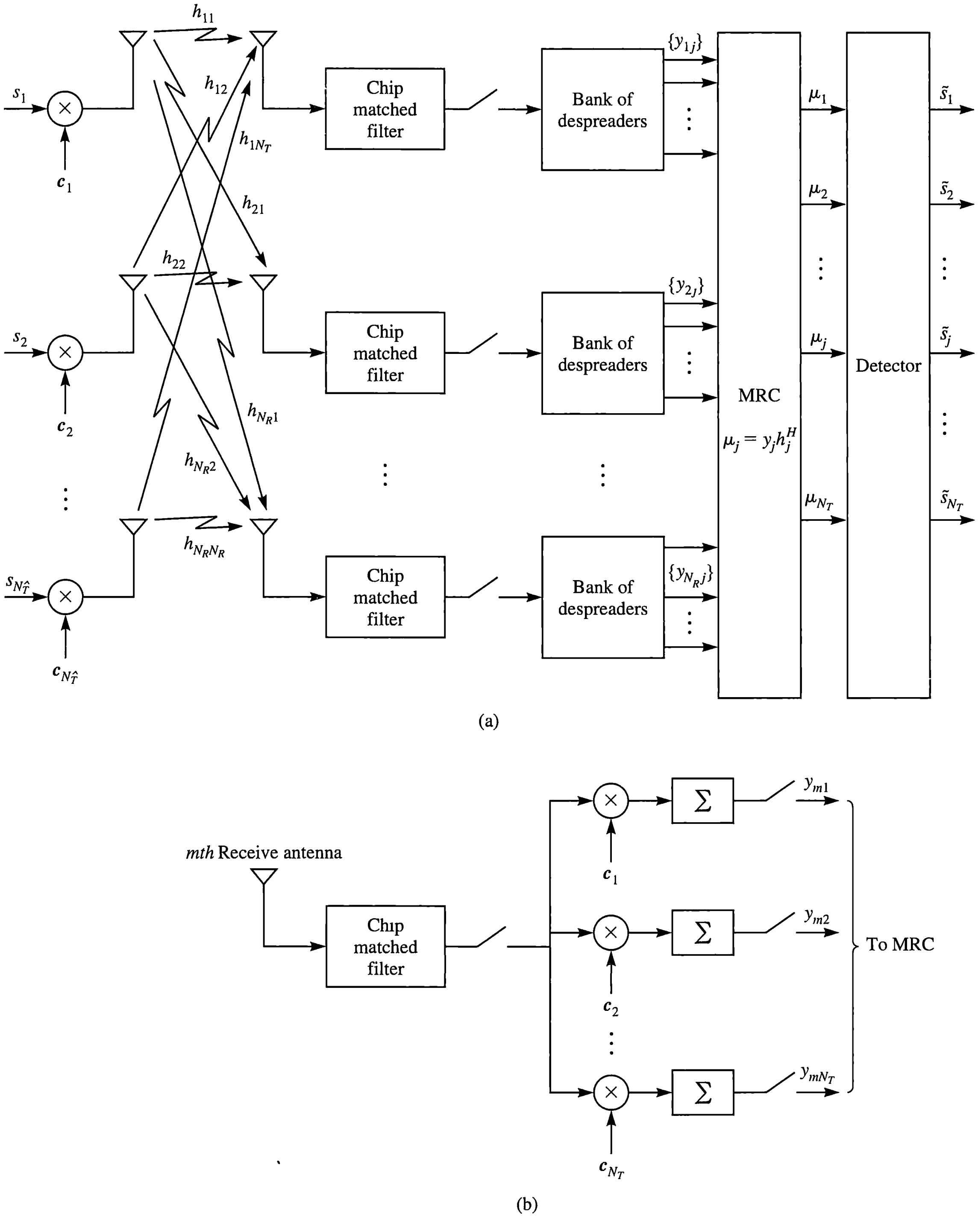

h_{11}
h_{12}
h_{1N_T}
h_{21}
h_{22}
h_{N_R1}
h_{N_R2}
$h_{N_RN_R}$
s_1
c_1
s_2
c_2
s_{N_T}
c_{N_T}
Chip matched filter
Chip matched filter
Chip matched filter
Bank of despreaders
Bank of despreaders
Bank of despreaders
$\{y_{1j}\}$
$\{y_{2j}\}$
$\{y_{N_Rj}\}$
MRC
$\mu_j = y_j h_j^H$
μ_1
μ_2
μ_j
μ_{N_T}
Detector
$\tilde{s}_1$
$\tilde{s}_2$
$\tilde{s}_j$
$\tilde{s}_{N_T}$
(a)
mth Receive antenna
Chip matched filter
Σ
Σ
Σ
c_1
c_2
c_{N_T}
y_{m1}
y_{m2}
y_{mN_T}
To MRC
(b)

FIGURE 15.3–1
MIMO system with spread spectrum signals.

its sampled output is fed to a bank of N_T correlators whose outputs are sampled at the end of each signaling interval, as illustrated in Figure 15.3–1(b). Since the spreading sequences are orthogonal, the N_T correlator outputs at the mth receive antenna are simply expressed as

$$y_{mj} = s_j \sqrt{\frac{\mathcal{E}_s}{N_T}} h_{mj} + \eta_{mj}, \qquad m = 1, 2, \ldots, N_R; \;\; j = 1, 2, \ldots, N_T \tag{15.3–3}$$

where $\{\eta_{mj}\}$ denote the additive noise components, which are assumed to be zero mean, complex-valued circularly symmetric Gaussian iid with variance $E\left[|\eta_{mj}|^2\right] = \sigma^2$.

It is convenient to express the N_R correlator outputs corresponding to the same transmitted symbol s_j in vector form as

$$\boldsymbol{y}_j = \sqrt{\frac{\mathcal{E}_s}{N_T}} s_j \boldsymbol{h}_j + \boldsymbol{\eta}_j \tag{15.3–4}$$

where $\boldsymbol{y}_j = [y_{1j}\, y_{2j} \cdots y_{N_R j}]^t$, $\boldsymbol{h}_j = [h_{1j}\, h_{2j} \cdots h_{N_R j}]^t$, and $\boldsymbol{\eta}_j = [\eta_{1j}\, \eta_{2j} \cdots \eta_{N_R j}]^t$. The optimum combiner is a maximal ratio combiner (MRC) for each of the transmitted symbols $\{s_j\}$. Thus, the output of the MRC for the jth signal is

$$\begin{aligned} \mu_j &= \boldsymbol{h}_j^H \boldsymbol{y}_j \\ &= \sqrt{\frac{\mathcal{E}_s}{N_T}} s_j \, \| \boldsymbol{h}_j \|_F^2 + \boldsymbol{h}_j^H \boldsymbol{\eta}_j, \qquad j = 1, 2, \ldots, N_T \end{aligned} \tag{15.3–5}$$

The decision metrics $\{\mu_j\}$ are the inputs to the detector, which makes an independent decision on each symbol in the set $\{s_j\}$ of transmitted symbols.

We observe that the use of orthogonal spreading sequences in a MIMO system transmitting over a frequency-nonselective channel significantly simplifies the detector and, for a spatially white channel, yields N_R-order diversity for each of the transmitted symbols $\{s_j\}$. The evaluation of the error rate performance of the detector for standard signal constellations such as PSK and QAM is relatively straightforward.

Frequency-Selective Channel If the channel is frequency-selective, the orthogonality property of the spreading sequences no longer holds at the receiver. That is, the channel multipath results in multiple received signal components which are offset in time. Consequently, the correlator outputs at each of the antennas contain the desired symbol plus the other $N_T - 1$ transmitted symbols, each scaled by the corresponding cross-correlations between pairs of sequences. Due to the presence of intersymbol interference, the MRC is no longer optimum. Instead, the optimum detector is a joint maximum-likelihood detector for the N_T transmitted symbols received at the N_R receive antennas.

In general, the implementation complexity of the optimum detector in a frequency-selective channel is extremely high. In such channels, a suboptimum receiver may be employed. A receiver structure that is readily implemented in a MIMO frequency-selective channel employs adaptive equalizers at each of the N_R receivers prior to despreading the spread spectrum signals. Figure 15.3–2 illustrates the basic receiver

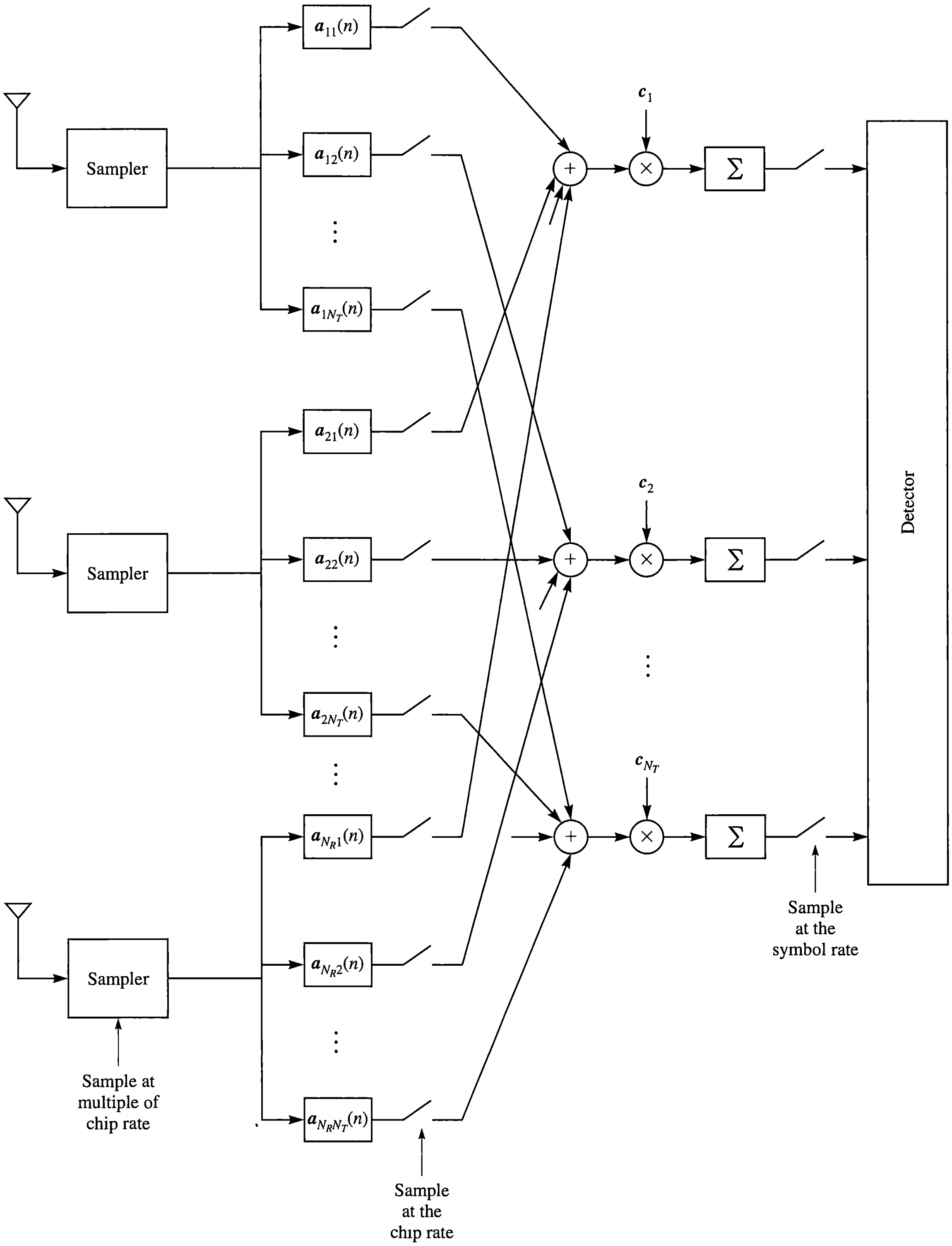

$a_{11}(n)$
$a_{12}(n)$
$a_{1N_T}(n)$
$a_{21}(n)$
$a_{22}(n)$
$a_{2N_T}(n)$
$a_{N_R1}(n)$
$a_{N_R2}(n)$
$a_{N_RN_T}(n)$
Sampler
Sampler
Sampler
Sample at multiple of chip rate
Sample at the chip rate
c_1
c_2
c_{N_T}
+
×
Σ
Detector
Sample at the symbol rate

FIGURE 15.3–2
A MIMO receiver structure for a frequency-selective channel.

structure. The received signal at each receive antenna is sampled at some multiple of the chip rate and fed to a parallel bank of N_T fractionally spaced linear equalizers, whose outputs are sampled at the chip rate. After combining the respective N_R equalizer outputs, the N_T signals are despread and fed to the detector, as illustrated in Figure 15.3–2. Alternatively DFEs may be used, where the feedback filters are operated at the symbol rate.

Training signals for the equalizers may be provided to the receiver by transmitting a pilot signal from each transmit antenna. These pilot signals may be spread spectrum signals that are simultaneously transmitted along with the information-bearing signals. Using the pilot signals, the equalizer coefficients can be adjusted recursively by employing a LMS- or RLS-type algorithm.

15.3–2 Multiplexing Gain Versus Diversity Gain

As we have observed from our previous discussion, the use of orthogonal spreading sequences to transmit multiple data symbols makes it possible for the receiver to separate the data symbols by correlating the received signal with each of the spreading sequences. For example, let us consider the MISO system shown in Figure 15.3–3, which has N_T transmit antennas and one receive antenna. As shown, N_T different symbols are transmitted simultaneously on the N_T transmit antennas. The receiver employs a parallel

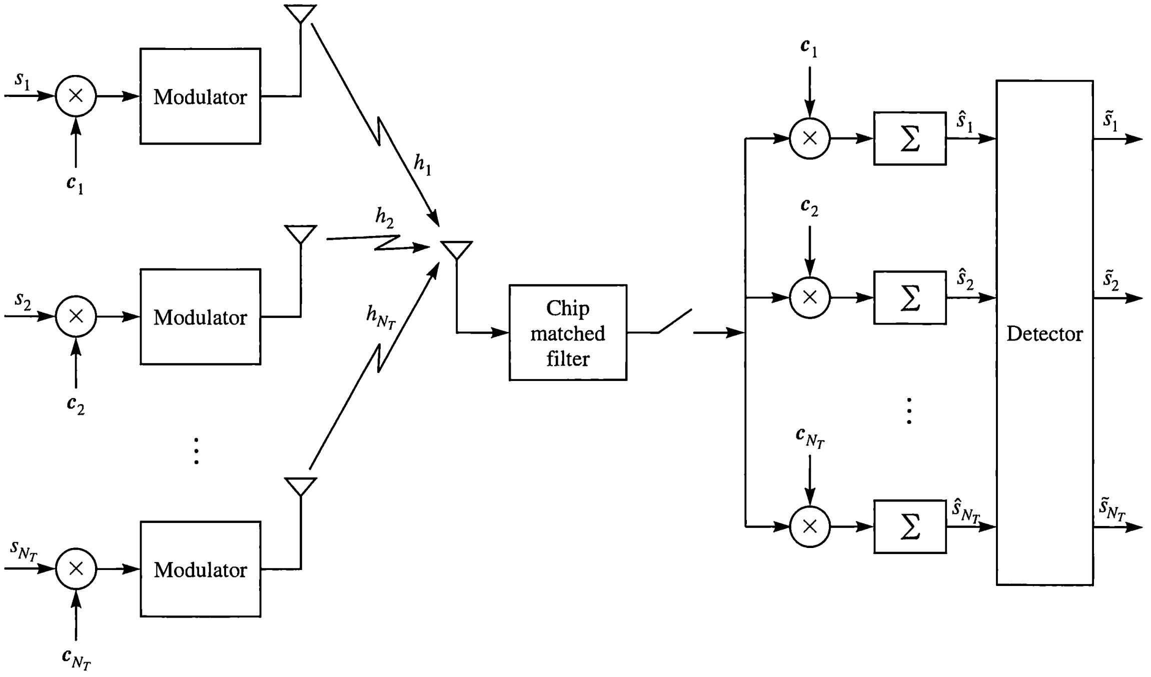


FIGURE 15.3–3
MISO system with spread spectrum signals.

bank of N_T correlators. Thus, the output of the jth correlator is

$$y_j = \sqrt{\frac{\mathcal{E}_s}{N_T}} s_j h_j + \eta_j, \qquad j = 1, 2, \ldots, N_T \tag{15.3–6}$$

where h_j is the complex-valued channel parameter associated with the propagation of the jth transmitted signal. Hence, the detector computes the decision variables $\{y_j h_j^*, j = 1, 2, \ldots, N_T\}$ and makes an independent decision on each transmitted symbol. In this configuration, the MISO system achieves a multiplexing gain (increase in data rate) of N_T, but there is no diversity gain. Alternatively, if two or more transmitting antennas transmit the same information symbol, the receiver can employ a maximal ratio combiner to combine the received signals carrying the same information and, thus, achieve an order of diversity of 2 or more at the expense of reducing the multiplexing gain. If all N_T transmit antennas are used to transmit the same information symbol, the receiver can achieve N_T-order diversity, but there would be no multiplexing gain. Thus, we observe that there is a tradeoff between muliplexing gain and diversity gain.

More generally, in a MIMO system with N_T transmit antennas and N_R receive antennas, the multiplexing gain can vary from 1 to N_T and the diversity gain can vary from $N_R N_T$ to N_R, respectively. Thus, an increase in diversity gain is offset by a corresponding decrease in multiplexing gain and vice versa. Although we have described this tradeoff between multiplexing gain and diversity gain in the context of orthogonal spreading sequences, this tradeoff is also appropriate in the context of narrowband signals.

15.3–3 Multicode MIMO Systems

In Sections 15.3–1 and 15.3–2, we considered spread spectrum MIMO systems in which a single sequence was used at each transmitting antenna to spread a single information symbol. However, it is possible to employ multiple orthogonal sequences at each transmitting antenna, to transmit multiple information symbols and thus to increase the data rate.

Figure 15.3–4 illustrates this concept with the use of two transmit and two receive antennas ($N_R = N_T = 2$). There are K orthogonal spreading sequences that are used to spread the spectrum of K information symbols at each transmitter. The same K spreading sequences are used at all the transmitters. Thus, with N_T transmit antennas there are $K N_T$ information symbols that are transmitted simultaneously. At each transmitter, the sum of K spread signals is multiplied by a pseudorandom sequence $\boldsymbol{p}_j$, called a *scrambling sequence*, consisting of statistically independent, equally probable $+1$s and -1s occurring at the chip rate of the orthogonal sequences $\{\boldsymbol{c}_k\}$. The scrambling sequences used at the N_T different transmitters are assumed to be statistically independent. These scrambling sequences serve as a means to separate (orthogonalize) the transmissions among the N_T transmit antennas, and have a length L_s, which may be equal to or larger than the length L_c of the orthogonal sequences, where L_c is the number of chips per information symbol. The scrambled orthogonal signals at each

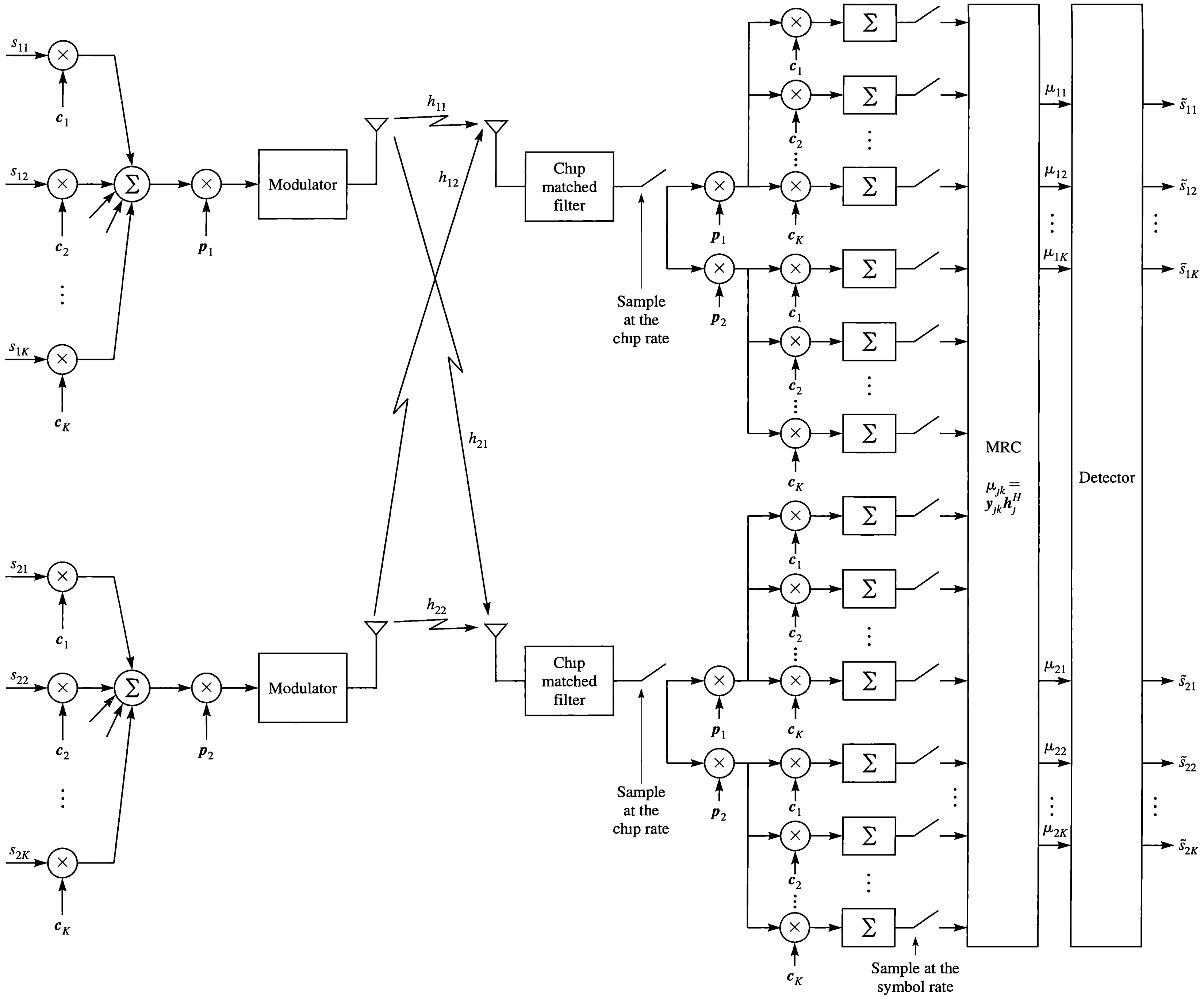

s_{11}
s_{12}
s_{1K}
c_1
c_2
c_K
p_1
Modulator
s_{21}
s_{22}
s_{2K}
c_1
c_2
c_K
p_2
Modulator
h_{11}
h_{12}
h_{21}
h_{22}
Chip matched filter
Sample at the chip rate
Chip matched filter
Sample at the chip rate
p_1
p_2
p_1
p_2
c_1
c_2
c_K
Σ
Sample at the symbol rate
MRC
$\mu_{jk} = y_{jk} h_j^H$
μ_{11}
μ_{12}
μ_{1K}
μ_{21}
μ_{22}
μ_{2K}
Detector
$\tilde{s}_{11}$
$\tilde{s}_{12}$
$\tilde{s}_{1K}$
$\tilde{s}_{21}$
$\tilde{s}_{22}$
$\tilde{s}_{2K}$

FIGURE 15.3–4
Modulator and demodulator for a multicode MIMO system.

antenna may be expressed as

$$s_j(t) = \sqrt{\frac{\mathcal{E}_s}{KN_T}} \sum_{k=1}^{K} s_{jk} \sum_{i=1}^{L_c} c_{ki} p_{ji} g(t - iT_c), \qquad j = 1, 2, \ldots, N_T; 0 \le t \le T \tag{15.3–7}$$

where $\boldsymbol{p}_j$ is the scrambling sequence at the jth transmitter, $\boldsymbol{s}_j = [s_{j1}\, s_{j2} \cdots s_{jK}]^t$ is the vector of information symbols transmitted from the jth antenna, $\boldsymbol{c}_k = [c_{k1}\, c_{k2} \cdots c_{kL_c}]$ is the kth orthogonal spreading sequence, $g(t)$ is the chip signal pulse of duration T_c and energy $1/L_c$, and $\mathcal{E}_s/KN_T$ is the average energy per transmitted information symbol at each antenna.

At each receive antenna, the received signals are passed through a chip matched filter and sampled at the chip rate. The samples at the output of the chip matched filters are descrambled and cross-correlated with each of the K orthogonal sequences. The correlator outputs are sampled at the symbol rate. Assuming that the scrambling sequences are orthogonal, these samples may be expressed as

$$\boldsymbol{y}_{jk} = \sqrt{\frac{\mathcal{E}_s}{KN_T}} s_{jk} \boldsymbol{h}_j + \boldsymbol{\eta}_{jk}, \qquad j = 1, 2, \ldots, N_T; \; k = 1, 2, \ldots, K \tag{15.3–8}$$

where $\boldsymbol{y}_{jk} = [y_{1jk}\, y_{2jk} \cdots y_{N_R jk}]^t$, $\boldsymbol{h}_j = [h_{1j}\, h_{2j} \cdots h_{N_R j}]^t$, and $\boldsymbol{\eta}_{jk} = [\eta_{1jk}\eta_{2jk} \cdots \eta_{N_R jk}]^t$ is the additive Gaussian noise vector. Thus, the transmitted symbols are decoupled by use of orthogonal scrambling and spreading sequences. These samples are fed to the maximal ratio combiner which computes the metrics

$$\begin{aligned} \mu_{jk} &= \boldsymbol{h}_j^H \boldsymbol{y}_{jk} \\ &= \sqrt{\frac{\mathcal{E}_s}{KN_T}} s_{jk} \, \| \boldsymbol{h}_j \|_F^2 + \boldsymbol{h}_j^H \, \boldsymbol{\eta}_{jk}, \qquad j = 1, 2, \ldots, N_T; \quad k = 1, 2, \ldots, K \end{aligned} \tag{15.3–9}$$

These metrics are passed to the detector which makes a decision on each of the transmitted information symbols based on a Euclidean distance criterion. We should note that if the scrambling sequences are not orthogonal, we have intersymbol interference among the symbols transmitted on the N_T antennas. In such a case, a multisymbol (or multiuser) detector must be employed.

In a frequency-selective channel, the orthogonality among the multiple codes is destroyed. In such channels, a practical implementation of the receiver employs adaptive equalizers to restore the orthogonality of the codes and mitigates the effects of interchip and intersymbol interference. Figure 15.3–5 illustrates such a receiver structure. Training signals for the equalizers are usually provided to the receiver by transmitting a pilot signal from each transmit antenna. These pilot signals may be spread spectrum signals that are simultaneously transmitted along with the information-bearing signals. For example, the pilot signals may be transmitted with the spreading code $\boldsymbol{c}_1$ at each transmit antenna. Using the pilot signals, the equalizer coefficients can be adjusted recursively by employing either an LMS or RLS type of algorithm.

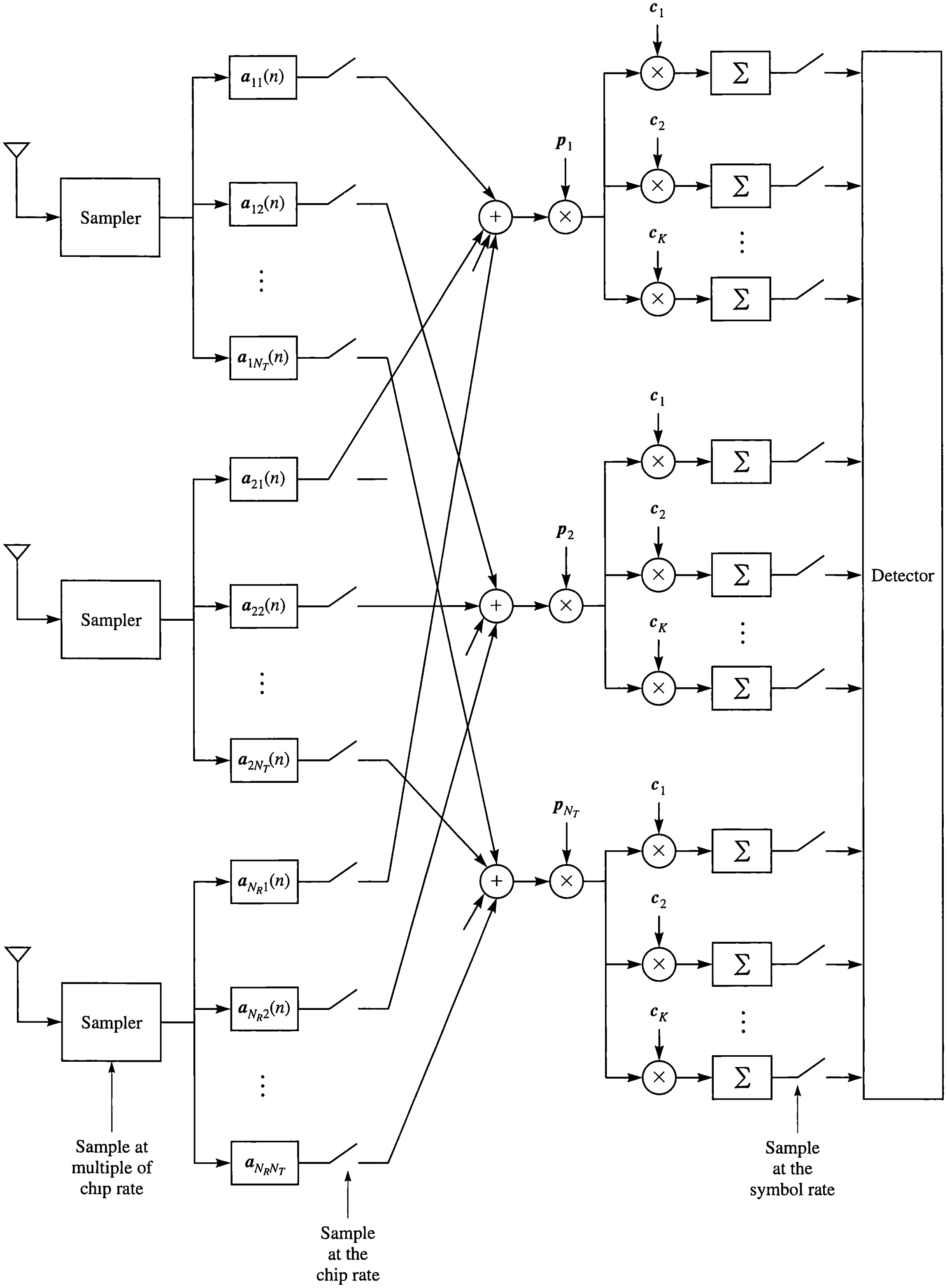

Sampler
Sampler
Sampler
$a_{11}(n)$
$a_{12}(n)$
$a_{1N_T}(n)$
$a_{21}(n)$
$a_{22}(n)$
$a_{2N_T}(n)$
$a_{N_R1}(n)$
$a_{N_R2}(n)$
$a_{N_RN_T}$
p_1
p_2
p_{N_T}
c_1
c_2
c_K
+
×
Σ
Detector
Sample at multiple of chip rate
Sample at the chip rate
Sample at the symbol rate

FIGURE 15.3–5
Receiver structure for a MIMO multicode system in a frequency-selective MIMO channel.

15.4

CODING FOR MIMO CHANNELS

In this section we describe two different approaches to code design for MIMO channels and evaluate their performance for frequency-nonselective Rayleigh fading channels. The first approach is based on using conventional block or convolutional codes with interleaving to achieve signal diversity. The second approach is based on code design that is tailored for multiple-antenna systems. The resulting codes are called *space-time codes*. We begin by recapping the error rate performance of coded SISO systems in Rayleigh fading channels.

15.4–1 Performance of Temporally Coded SISO Systems in Rayleigh Fading Channels

Let us consider a SISO system, as shown in Figure 15.4–1, where the fading channel is frequency-nonselective and the fading process is Rayleigh-distributed. The encoder generates either an (n, k) linear binary block code or an (n, k) binary convolutional code. The interleaver is assumed to be sufficiently long that the transmitted signals conveying the coded bits fade independently. The modulation is binary PSK, DPSK, or FSK.

The error probabilities for the coded SISO channel with Rayleigh fading are given in Sections 14.4 and 14.7. Let us consider linear block codes first. From Section 7.2–4, the union bound on the codeword error probability for soft decision decoding is

$$P_e < \sum_{m=2}^{M} P_2(w_m) < (M-1)P_2(d_{\min}) < 2^k P_2(d_{\min}) \tag{15.4–1}$$

where $P_2(w_m)$ is the pairwise error probability given by the expression (see Section 14.7–1)

$$P_2(w_m) = \left(\frac{1-\psi}{2}\right)^{w_m} \sum_{k=0}^{w_{m-1}} \binom{w_m - 1 + k}{k} \left(\frac{1+\psi}{2}\right)^k \tag{15.4–2}$$

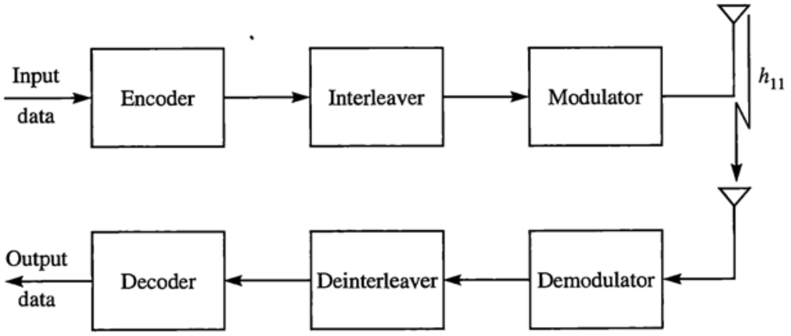


FIGURE 15.4–1
Temporally coded SISO system.

and

$$\psi = \begin{cases} \sqrt{\dfrac{\bar{\gamma}_b R_c}{1+\bar{\gamma}_b R_c}} & \text{BPSK} \\ \bar{\gamma}_b R_c/(1+\bar{\gamma}_b R_c) & \text{DPSK} \\ \bar{\gamma}_b R_c/(2+\bar{\gamma}_b R_c) & \text{FSK (noncoherent detection)} \end{cases} \tag{15.4–3}$$

For simplicity, we will use the simpler (looser) upper bound obtained by assuming that $\bar{\gamma}_b \gg 1$ in the expression for $P_2(d_{\min})$. Thus, we obtain

$$\begin{aligned} P_e &< 2^k P_2(d_{\min}) \\ &< 2^k \binom{2d_{\min}-1}{d_{\min}} \left(\frac{1}{qR_c\bar{\gamma}_b}\right)^{d_{\min}} \end{aligned} \tag{15.4–4}$$

where

$$q = \begin{cases} 4 & \text{BPSK} \\ 2 & \text{DPSK} \\ 1 & \text{FSK (noncoherent detection)} \end{cases} \tag{15.4–5}$$

We observe that for soft decision decoding, the error probability decays exponentially as $1/\bar{\gamma}_b R_c$, where the exponent is equal to $d_{\min}$, the minimum Hamming distance of the block codes.

For hard decision decoding, we employ the Chernov bound given in Section 14.4, which may be expressed as

$$P_e < 2^k[4p(1-p)]^{d_{\min}/2} \tag{15.4–6}$$

where the error probability per coded bit is given as

$$p = \frac{1-\psi}{2} \tag{15.4–7}$$

and ψ is defined in Equation 15.4–3. For $\bar{\gamma}_b \gg 1$, the Chernov bound simplifies to

$$P_e < 2^k \left(\frac{4}{qR_c\bar{\gamma}_b}\right)^{d_{\min}/2} \tag{15.4–8}$$

where q is defined in Equation 15.4–5. As in the case of soft decision decoding, the error probability decays exponentially as $1/\bar{\gamma}_b R_c$; however, the exponent for hard decision decoding is $d_{\min}/2$. Therefore, soft decision decoding provides twice the signal diversity that is obtained by hard decision decoding.

For convolutional codes with soft decision decoding, we use the union bound derived in Section 14.3, namely,

$$P_b < \sum_{d=d_{\text{free}}}^{\infty} \beta_d P_2(d) \tag{15.4–9}$$

where $P_2(d)$ is given by Equation 15.4–2 and ψ is defined by Equation 15.4–3. If $\bar{\gamma}_b \gg 1$, we obtain the simpler form for the pairwise error probability, i.e.,

$$P_2(d) \approx \binom{2d-1}{d} \left(\frac{1}{qR_c\bar{\gamma}_b} \right)^d \tag{15.4–10}$$

where q is defined by Equation 15.4–5. We observe that the leading term in Equation 15.4–9 has an exponent of $d = d_{\text{free}}$. Hence, for soft decision decoding, the leading term in the error probability decays exponentially as $1/\bar{\gamma}_b R_c$, where the exponent is d_{free}, the free distance of the convolutional code.

For hard decision decoding, we again use the Chernov bound for the pairwise error probability

$$P_2(d) < [4p(1-p)]^{d/2} \tag{15.4–11}$$

where p is defined by Equation 15.4–7 and ψ is defined by Equation 15.4–3. Hence, with $\bar{\gamma}_b \gg$, $P_2(d)$ simplifies to

$$P_2(d) < \left(\frac{4}{qR_c\bar{\gamma}_b} \right)^{d/2} \tag{15.4–12}$$

and the bit error probability is upper-bounded as

$$P_b < \sum_{d=d_{\text{free}}}^{\infty} \beta_d \left(\frac{4}{qR_c\bar{\gamma}_b} \right)^{d/2} \tag{15.4–13}$$

As in the case of block codes, we observe that with hard decision decoding, the signal diversity achieved by the code is reduced by a factor of 2 compared with soft decision decoding.

With this background on the performance of coded SISO systems, we now consider the performance of coded MIMO systems.

15.4–2 Bit-Interleaved Temporal Coding for MIMO Channels

We consider the MIMO system as shown in Figure 15.4–2, which has N_T transmit antennas and N_R receive antennas ($N_R \geq N_T$). The encoder may generate either a binary block code or a convolutional code. The interleaver is selected to be sufficiently long that the coded bits in a block of the block code or in several constraint lengths of the convolutional code fade independently. The MIMO channel is assumed to be frequency-nonselective with zero-mean, complex-valued, circularly symmetric Gaussian distributed coefficients $\{h_{ij}\}$, which are identically distributed and mutually statistically independent. The channel metrix $\boldsymbol{H}$ is assumed to have full rank.

The demodulator output in each signal interval is the vector $\boldsymbol{y}$ given by Equation 15.1–10. For hard decision decoding, the vector $\boldsymbol{y}$ is fed to the detector, which may employ any of the three detection algorithms (MLD, MMSE, ICD) described in Section 15.1–2 to make the hard decisions on the transmitted bits. For soft decision decoding, the vector $\boldsymbol{y}$, after deinterleaving, is fed to the decoder. Similarly, for hard

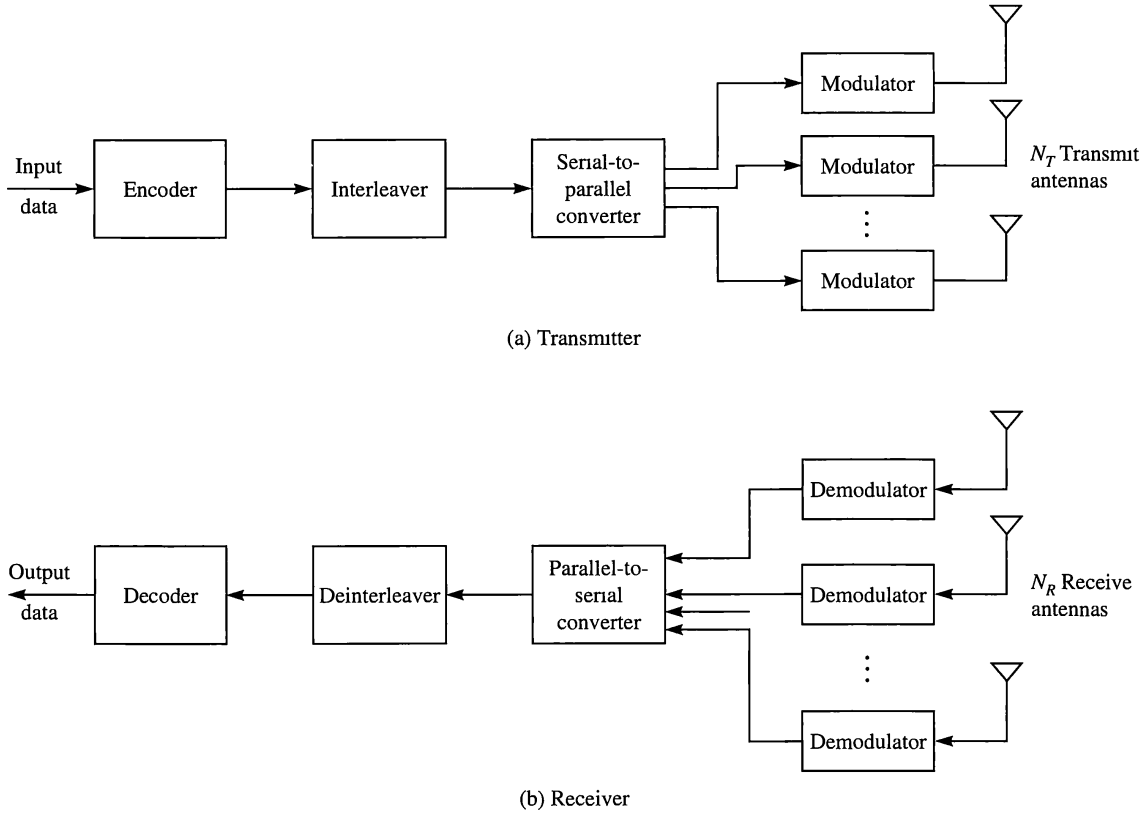

Input data
Encoder
Interleaver
Serial-to-parallel converter
Modulator
Modulator
Modulator
N_T Transmit antennas
(a) Transmitter
Output data
Decoder
Deinterleaver
Parallel-to-serial converter
Demodulator
Demodulator
Demodulator
N_R Receive antennas
(b) Receiver

FIGURE 15.4–2
Bit-interleaved temporally coded MIMO system.

decision decoding, the bits from the detector output are deinterleaved and fed to the decoder.

Let us consider the amount of signal diversity that is achieved in the MIMO system that employs spatial multiplexing of N_T. Recall from Section 15.1–2 that with hard decision detection in an uncoded system, we achieved $(N_R - N_T + 1)$-order signal diversity with linear detection and N_R-order signal diversity with the optimum maximum-likelihood detector (MLD). From our discussion in Section 15.4–1, we observed that the code provides a diversity of order $d_{\min}/2$ or $d_{\text{free}}/2$. Therefore, in a coded MIMO system, the total signal diversity achieved with a linear detector and a hard decision decoder is $(N_R - N_T + 1)d_{\min}/2$ or $(N_R - N_T + 1)d_{\text{free}}/2$. On the other hand, if soft decision decoding is employed, the total diversity order is $N_R d_{\min}$ or $N_R d_{\text{free}}$.

We demonstrate the additional diversity achieved with coding and bit-interleaving by computer simulation of the MIMO system shown in Figure 15.4–2 for a rate $R_c = 1/2$ convolutional code with $d_{\text{free}} = 5$ and BPSK modulation. Figures 15.4–3 and 15.4–4 illustrate the performance of the MIMO system for binary PSK with hard decision decoding and soft decision decoding, for $(N_T, N_R) = (2, 2)$ and $(N_T, N_R) = (2, 3)$. We observe that coding with interleaving improves the performance of the MIMO system relative to the performance of the uncoded system at the cost of a reduction in the data throughput rate by the reciprocal of the code rate. For $(N_T, N_R) = (2, 3)$ and hard decision decoding, the MMSE detector with coding performs almost as well as the MLD

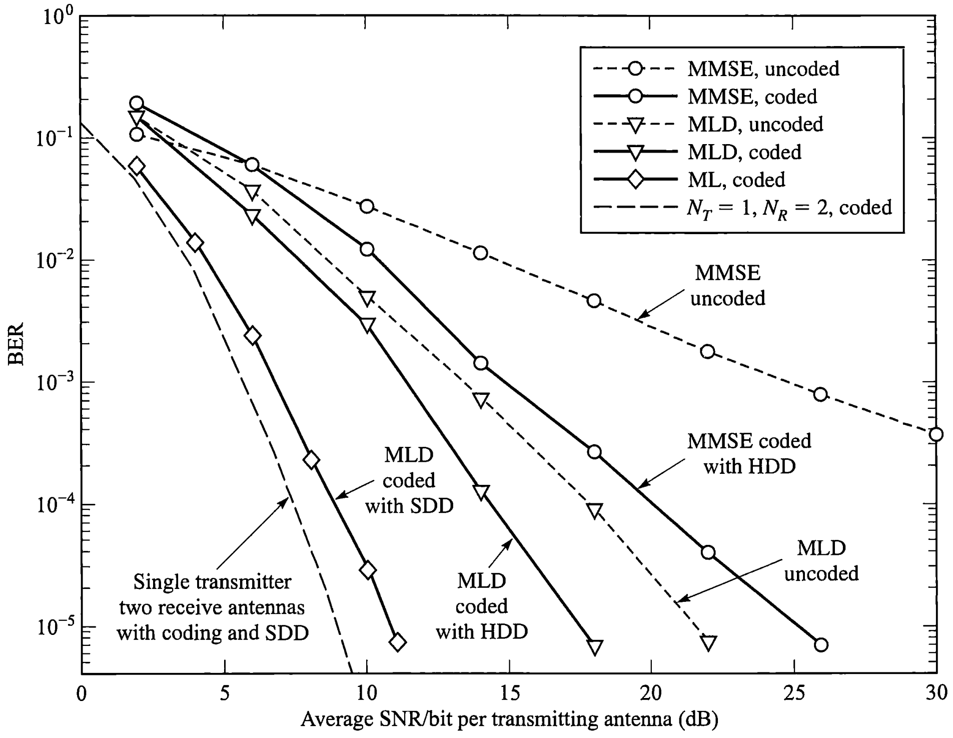


FIGURE 15.4–3
Performance of coded ($R_c = 1/2, d_{\text{free}} = 5$) systems with $N_T = N_R = 2$.

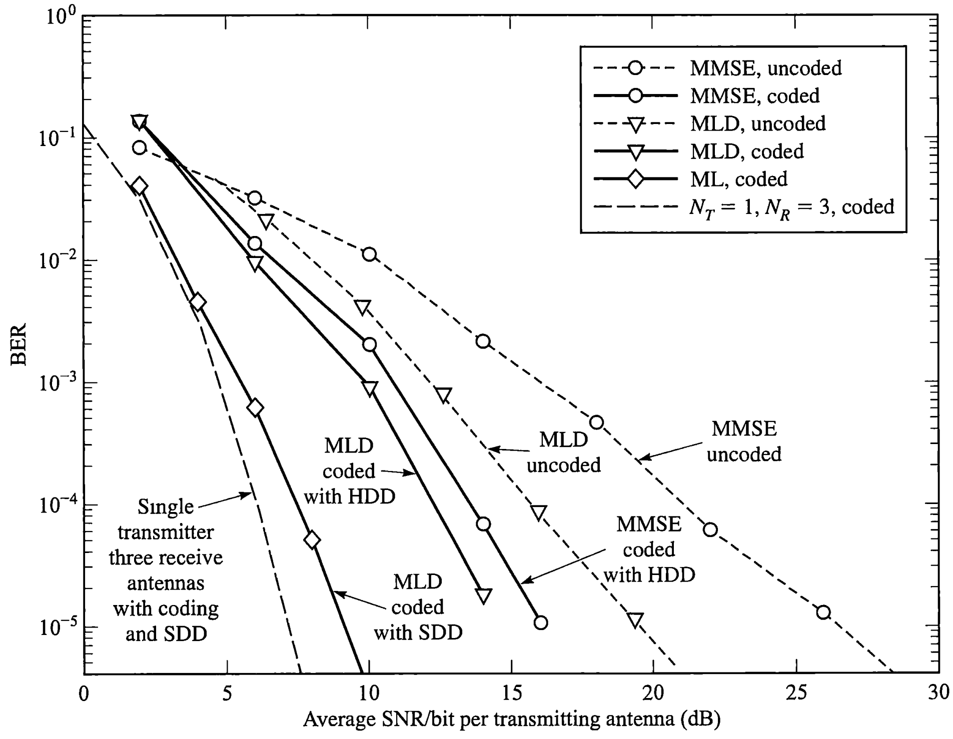


FIGURE 15.4–4
Performance of coded ($R_c = 1/2, d_{\text{free}} = 5$) systems with $N_T = 2, N_R = 3$.

detector with coding. In this case, the signal diversity provided by the convolutional code enhances the performance of the MMSE detected data more than the performance of the MLD detected data. We also observe that maximum-likelihood, soft decision decoding is significantly better than MLD with hard decision decoding. For example, at 10^{-5}, the difference in performance is more than 5 dB for $(N_T, N_R) = (2, 3)$. This performance advantage is due to the factor of 2 difference in the order of diversity achieved by the two types of decoders.

Also plotted in Figures 15.4–3 and 15.4–4 is the ideal performance of rate 1/2, $d_{\text{free}} = 5$ coded SIMO $(N_T, N_R) = (1, 2)$ and $(N_T, N_R) = (1, 3)$ systems. The signal diversity achieved by these two systems with soft decision decoding is 10 and 15, respectively. We observe that there is about a 2-dB degradation at $P_b = 10^{-5}$ in the performance of the soft decision decoded (2, 2) and (2, 3) MIMO systems compared to the ideal performance of the corresponding SIMO systems. This loss in performance is attributed to the interference resulting from the use of multiple transmitting antennas.

The simulation results shown in Figures 15.4–3 and 15.4–4 serve to reinforce our analytical results on the signal diversity provided by coding with bit interleaving in a MIMO system. The performance superiority of maximum-likelihood soft decision decoding over hard decision decoding is clearly evident in these simulation results.

In this section we employed a single encoder and a single interleaver to generate the coded symbols for transmission on the N_T antennas and a single deinterleaver and decoder at the receiver. An alternative approach that has been considered in the literature is to employ separate but identical encoding and interleaving on the dimultiplexed streams fed to each of the transmit antennas. This approach requires N_T parallel encoders and interleavers at the transmitter and N_T parallel decoders and deinterleavers at the receiver. It is especially suitable for situations where multiple data streams from different users are to be transmitted in parallel on multiple transmit antennas.

15.4–3 Space-Time Block Codes for MIMO Channels

Let us now consider the MIMO system illustrated in Figure 15.4–5. At the transmitter, the sequence of information bits is fed to a block encoder that maps a block of bits into signal points selected from a signal constellation such as PAM, PSK, or QAM, consisting of $M = 2^b$ signal points. The signal points generated by the encoder as a block are fed to a parallel set of identical modulators which map the signal points into corresponding waveforms that are transmitted simultaneously on the N_T antennas.

A *space-time block code* (STBC) is defined by a generator matrix $\boldsymbol{G}$, having N rows and N_T columns, of the form

$$\boldsymbol{G} = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1N_T} \\ g_{21} & g_{22} & \cdots & g_{2N_T} \\ \vdots & \vdots & & \vdots \\ g_{N1} & g_{N2} & \cdots & g_{NN_T} \end{bmatrix} \tag{15.4–14}$$

in which the elements $\{g_{ij}\}$ are signal points resulting from a mapping of information bits to corresponding signal points from a binary or M-ary signal constellation. By

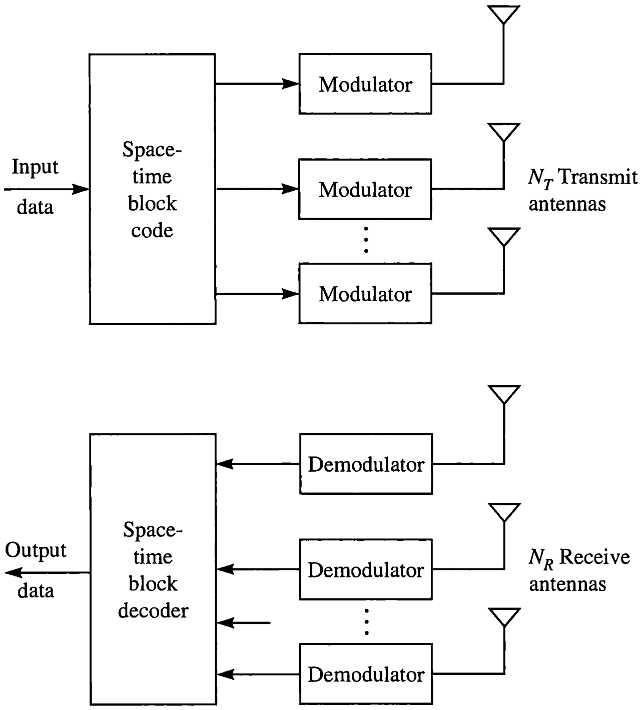


FIGURE 15.4–5
Space-time block coded MIMO system.

employing N_T transmit antennas, each row of $\boldsymbol{G}$ consisting of N_T signal points (symbols) is transmitted on the N_T antennas in a time slot. Thus, the first row of N_T symbols is transmitted on the N_T antennas in the first time slot, the second row of N_T symbols is transmitted on the N_T antennas in the second time slot, and the Nth row of N_T symbols is transmitted on the N_T antennas in the Nth time slot. Therefore, N time slots are used to transmit the symbols in the N rows of the generator matrix $\boldsymbol{G}$.

In the design of the generator matrix of a STBC, it is desirable to focus on three principal objectives: (1) achieving the highest possible diversity of $N_T N_R$, (2) achieving the highest possible spatial rate, and (3) minimizing the complexity of the decoder. Our treatment considers these three objectives.

The Alamouti STBC

Alamouti (1998) devised a STBC for $N_T = 2$ transmit antennas and $N_R = 1$ receive antenna. The generator matrix for the Alamouti code is given as

$$\boldsymbol{G} = \begin{bmatrix} s_1 & s_2 \\ -s_2^* & s_1^* \end{bmatrix} \tag{15.4–15}$$

where s_1 and s_2 are two signal points selected from an M-ary PAM, or PSK or QAM signal constellation with $M = 2^b$ signal points. Thus, $2b$ data bits are mapped into two signal points (symbols) s_1 and s_2 from the M-ary signal constellation. The symbols s_1 and s_2 are transmitted on the two antennas in the first time slot, and the symbols $-s_2^*$ and s_1^* are transmitted on the two antennas in the second time slot. Thus, two symbols, s_1 and s_2, are transmitted in two time slots. Consequently, the *spatial code rate* $R_s = 1$ for the Alamouti code. This is the highest possible rate for a (orthogonal) STBC.

The MISO channel matrix for the $N_T = 2$, $N_R = 1$ channel, based on a frequency-nonselective model, is

$$\boldsymbol{H} = [h_{11}\, h_{12}] \tag{15.4–16}$$

In the decoding of the STBC, we assume that $\boldsymbol{H}$ is constant over the two time slots. Consequently, the signal at the output of the matched filter demodulator of the receiver in the two time slots is

$$\begin{aligned} y_1 &= h_{11}s_1 + h_{12}s_2 + \eta_1 \\ y_2 &= -h_{11}s_2^* + h_{12}s_1^* + \eta_2 \end{aligned} \tag{15.4–17}$$

where η_1 and η_2 are zero-mean, circularly symmetric complex-valued uncorrelated Gaussian random variables with equal variance σ_η^2.

Let us consider ML decoding of the symbols in Equation 15.4–17, with the objective of achieving the full diversity of the STBC. Since η_1 and η_2 are uncorrelated zero-mean Gaussian random variables with equal variance, the joint conditional PDF of y_1 and y_2 is

$$\begin{aligned} p(y_1, y_2 | h_{11}, h_{12}, s_1, s_2) = \frac{1}{2\pi\sigma_\eta^2} \exp\left\{-\left[|y_1 - h_{11}s_1 - h_{12}s_2|^2\right.\right. \\ \left.\left. + |y_2 + h_{11}s_2^* - h_{12}s_1^*|^2\right]\right\}/2\sigma_\eta^2 \end{aligned} \tag{15.4–18}$$

Therefore, the Euclidean distance metric for ML decoding is

$$\mu(s_1, s_2) = |y_1 - h_{11}s_1 - h_{12}s_2|^2 + |y_2 + h_{11}s_2^* - h_{12}s_1^*|^2 \tag{15.4–19}$$

The optimum ML decoder computes the Euclidean metrics $\mu(s_1, s_2)$ for each possible pair of symbols and selects the symbol pair that results in the smallest metric.

The computational complexity of the ML decoding procedure is exponential in the number of symbol pairs; i.e., there are $M^2 = 2^{2b}$ symbol pairs in the above metric computations. However, the computational complexity can be reduced if we expand the right-hand side of Equation 15.4–19 and drop the term $|y_1|^2 + |y_2|^2$, which is irrelevant to the decision. Thus, we obtain

$$\begin{aligned} \mu(s_1, s_2) &= |s_1|^2 \left[|h_{11}|^2 + |h_{12}|^2\right] - 2\,\mathrm{Re}\left[y_1^* h_{11}s_1 + y_2 h_{12}^* s_1\right] \\ &\quad + |s_2|^2 \left[|h_{11}|^2 + |h_{12}|^2\right] - 2\,\mathrm{Re}\left[y_1^* h_{12}s_2 - y_2 h_{11}^* s_2\right] \\ &= \mu(s_1) + \mu(s_2) \end{aligned} \tag{15.4–20}$$

Now, we observe that the metrics $\mu(s_1)$ and $\mu(s_2)$ can be computed separately; i.e., we determine the symbol s_1 that minimizes $\mu(s_1)$ and the symbol s_2 that minimizes $\mu(s_2)$. Thus, the computational complexity is significantly reduced from computing M^2 metrics to $2M$ metrics.

A further simplification in decoding results when the signal points in the constellation have equal energy, as in PSK constellations. In such a case, the bias energy terms $|s_1|^2 \left[|h_{11}|^2 + |h_{12}|^2\right]$ and $|s_2|^2 \left[|h_{11}|^2 + |h_{12}|^2\right]$ can be ignored. Furthermore, the metrics $\mu(s_1)$ and $\mu(s_2)$ can be rearranged as correlation metrics, defined as

$$\begin{aligned} \mu_c(s_1) &= \mathrm{Re}\left[y_1^* h_{11}s_1 + y_2 h_{12}^* s_1\right] \\ \mu_c(s_2) &= \mathrm{Re}\left[y_1^* h_{12}s_2 - y_2 h_{11}^* s_2\right] \end{aligned} \tag{15.4–21}$$

That is, we correlate y_1^* with all possible values of s_1, scaled by h_{11}, and y_2 with all possible values of s_1, scaled by h_{12}^*, and select the s_1 that results in the largest correlation metric $\mu_c(s_1)$. A similar computation is performed to find the value of s_2 that yields the largest $\mu_c(s_2)$.

For PAM and QAM signal constellations, the correlation metrics include the bias terms in Equation 15.4–20. Hence, the correlation metrics may be expressed as

$$\begin{aligned}\mu_c(s_1) &= 2\,\mathrm{Re}\left[y_1^* h_{11} s_1 + y_2 h_{12}^* s_1\right] - |s_1|^2[|h_{11}|^2 + |h_{12}|^2]\\ \mu_c(s_2) &= 2\,\mathrm{Re}\left[y_1^* h_{12} s_2 - y_2 h_{11}^* s_2\right] - |s_2|^2[|h_{11}|^2 + |h_{12}|^2]\end{aligned} \tag{15.4–22}$$

It is interesting to note that for the particular symbol s_1 that is contained in y_1 and y_2, the signal component in the metric $\mu_c(s_1)$ is the largest possible and has the value

$$E[\mu_c(s_1)] = |s_1|^2\left[|h_{11}|^2 + |h_{12}|^2\right] \tag{15.4–23}$$

where the expectation is taken over the additive Gaussian noise. Similarly, we have

$$E[\mu_c(s_2)] = |s_2|^2\left[|h_{11}|^2 + |h_{12}|^2\right] \tag{15.4–24}$$

Since each signal term contains the term $\left[|h_{11}|^2 + |h_{12}|^2\right]$, the ML decoder achieves a diversity of order 2, which is the maximum possible diversity with $N_T = 2$ and $N_R = 1$ antennas.

Instead of computing the correlation metrics as defined in Equation 15.4–22, an equivalent detector (see Problem 15.15) computes the estimates of the symbols s_1 and s_2 as follows:

$$\begin{aligned}\hat{s}_1 &= y_1 h_{11}^* + y_2^* h_{12}\\ \hat{s}_2 &= y_1 h_{12}^* - y_2^* h_{11}\end{aligned} \tag{15.4–25}$$

and it selects the symbols $\tilde{s}_1$ and $\tilde{s}_2$ that are closest to $\hat{s}_1$ and $\hat{s}_2$ in Euclidean distance.

We make the following observation on the Alamouti STBC. First, we observe that the code achieves the largest possible diversity. Second, through the separation of the detector metrics given in Equation 15.4–22 or, equivalently, the estimates $\hat{s}_1$ and $\hat{s}_2$ given in Equation 15.4–25, the maximum-likelihood detector has low complexity. These two desirable properties were achieved as a result of the orthogonality characteristic of the generator matrix $\boldsymbol{G}$ for the Alamouti code, which we may express as

$$\boldsymbol{G} = \begin{bmatrix} g_1 & g_2 \\ -g_2^* & g_1^* \end{bmatrix} \tag{15.4–26}$$

We observe that the column vectors $\boldsymbol{v}_1 = (g_1, -g_2^*)^t$ and $\boldsymbol{v}_2 = (g_2, g_1^*)^t$ are orthogonal; i.e., $\boldsymbol{v}_1 \cdot \boldsymbol{v}_2^H = 0$ and, furthermore,

$$\boldsymbol{G}^H\boldsymbol{G} = \left[|g_1|^2 + |g_2|^2\right]\boldsymbol{I}_2 \tag{15.4–27}$$

where $\boldsymbol{I}_2$ is a 2×2 identity matrix. As a consequence of this property, when we express the received signal given in Equation 15.4–17 as

$$\begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{12}^* & -h_{11}^* \end{bmatrix}\begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} \eta_1 \\ \eta_2^* \end{bmatrix}$$

$$\boldsymbol{y} = \boldsymbol{H}_{21}\boldsymbol{s} + \boldsymbol{\eta} \tag{15.4–28}$$

and form the estimates $\hat{s}_1$ and $\hat{s}_2$ as prescribed in Equation 15.4–25 from $\boldsymbol{y}$ in Equation 15.4–28, we obtain

$$\begin{aligned}\begin{bmatrix}\hat{s}_1\\ \hat{s}_2\end{bmatrix} &= \begin{bmatrix}h_{11}^* & h_{12}\\ h_{12}^* & -h_{11}\end{bmatrix}\begin{bmatrix}y_1\\ y_2^*\end{bmatrix}\\ &= \boldsymbol{H}_{21}^H\boldsymbol{H}_{21}\boldsymbol{s} + \boldsymbol{H}_{21}^H\boldsymbol{\eta}\\ &= \left[|h_{11}|^2 + |h_{12}|^2\right]\boldsymbol{s} + \boldsymbol{H}_{21}^H\boldsymbol{\eta}\end{aligned} \tag{15.4–29}$$

Therefore,

$$\boldsymbol{H}_{21}^H\boldsymbol{H}_{21} = \left[|h_{11}|^2 + |h_{12}|^2\right]\boldsymbol{I}_2 \tag{15.4–30}$$

Thus, full diversity and low decoding complexity are achieved as a consequence of the orthogonality property of $\boldsymbol{G}$ given in Equation 15.4–27.

Alamouti Code with Multiple Receive Antennas

We shall now demonstrate that the Alamouti code achieves the maximum possible diversity of $N_T N_R = 2N_R$ when the number of receive antennas is increased to N_R. In this case, the $N_R \times 2$ channel matrix is

$$\boldsymbol{H} = [\boldsymbol{h}_1\,\boldsymbol{h}_2] = \begin{bmatrix}h_{11} & h_{12}\\ h_{21} & h_{22}\\ \vdots & \vdots\\ h_{N_R1} & h_{N_R2}\end{bmatrix} \tag{15.4–31}$$

In the first transmission, the received signal is

$$\boldsymbol{y}_1 = \boldsymbol{H}\begin{bmatrix}s_1\\ s_2\end{bmatrix} + \boldsymbol{\eta}_1 \tag{15.4–32}$$

and in the second transmission, the received signal is

$$\boldsymbol{y}_2 = \boldsymbol{H}\begin{bmatrix}-s_2^*\\ s_1^*\end{bmatrix} + \boldsymbol{\eta}_2 \tag{15.4–33}$$

As in the case of the MISO $N_T = 2, N_R = 1$ system, we may combine Equations 15.4–32 and 15.4–33 into the equation

$$\begin{bmatrix}\boldsymbol{y}_1\\ \boldsymbol{y}_2^*\end{bmatrix} = \boldsymbol{H}_{2N_R}\begin{bmatrix}s_1\\ s_2\end{bmatrix} + \begin{bmatrix}\boldsymbol{\eta}_1\\ \boldsymbol{\eta}_2^*\end{bmatrix} \tag{15.4–34}$$

where $\boldsymbol{H}_{2N_R}$ is defined as follows:

$$\boldsymbol{H}_{2N_R} = \begin{bmatrix}\boldsymbol{h}_1 & \boldsymbol{h}_2\\ \boldsymbol{h}_2^* & -\boldsymbol{h}_1^*\end{bmatrix} \tag{15.4–35}$$

Here $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ are the column vectors of the channel matrix given in Equation 15.4–31.

Suppose we form the estimates $\hat{s}_1$ and $\hat{s}_2$ as

$$\begin{aligned}\begin{bmatrix}\hat{s}_1\\ \hat{s}_2\end{bmatrix} &= \boldsymbol{H}_{2N_R}^H \begin{bmatrix}y_1\\ y_2^*\end{bmatrix}\\ &= \boldsymbol{H}_{2N_R}^H \boldsymbol{H}_{2N_R} \begin{bmatrix}s_1\\ s_2\end{bmatrix} + \boldsymbol{H}_{2N_R}^H \begin{bmatrix}\eta_1\\ \eta_2^*\end{bmatrix}\end{aligned} \tag{15.4–36}$$

It is easily verified that

$$\begin{aligned}\boldsymbol{H}_{2N_R}^H \boldsymbol{H}_{2N_R} &= \left[\sum_{i=1}^{N_R} |h_{i1}|^2 + |h_{i2}|^2)\right] \boldsymbol{I}_2\\ &= \|\boldsymbol{H}\|_F^2\ \boldsymbol{I}_2\end{aligned} \tag{15.4–37}$$

Consequently, Equation 15.4–36 simplifies to

$$\begin{bmatrix}\hat{s}_1\\ \hat{s}_2\end{bmatrix} = \|\boldsymbol{H}\|_F^2 \begin{bmatrix}s_1\\ s_2\end{bmatrix} + \boldsymbol{H}_{2N_R}^H \begin{bmatrix}\eta_1\\ \eta_2^*\end{bmatrix} \tag{15.4–38}$$

We conclude that the Alamouti code achieves the full diversity of $2N_R$ available in the MIMO system with $N_T = 2$ transmit and N_R receive antennas. Furthermore, the maximum-likelihood decoder bases its decisions on the decoupled estimates $\hat{s}_1$ and $\hat{s}_2$ obtained from Equation 15.4–36 as

$$\begin{aligned}\hat{s}_1 &= \boldsymbol{h}_1^H \boldsymbol{y}_1 + \boldsymbol{y}_2^H \boldsymbol{h}_2\\ \hat{s}_2 &= \boldsymbol{h}_2^H \boldsymbol{y}_1 - \boldsymbol{y}_2^H \boldsymbol{h}_1\end{aligned} \tag{15.4–39}$$

Hence, implementation complexity of the detector is minimized.

Orthogonal Code Design for $N_T > 2$ Transmit Antennas

The design of orthogonal generator matrices for more than $N_T = 2$ transmit antennas has been extensively studied. Jafarkhani (2005) gives a comprehensive treatment on their construction based on early work by Hurwitz and Radon (1922) on the design of real orthogonal matrices. A real $N \times N$ matrix $\boldsymbol{G}$ with entries $g_1, -g_1, g_2, -g_2, \ldots, g_N, -g_N$, is said to be orthogonal if

$$\boldsymbol{G}^t \boldsymbol{G} = \left(\sum_{i=1}^{N} g_i^2\right) \boldsymbol{I}_N \tag{15.4–40}$$

where $\boldsymbol{I}_N$ is the $N \times N$ identity matrix. It can be shown (Jafarkhani (2005)) that rate $R_s = 1$ real orthogonal matrix designs exist only for $N = 2, 4, 8$. For example, a real orthogonal matrix for $N_T = 4$ transmit antennas is the following:

$$\boldsymbol{G} = \begin{bmatrix} g_1 & g_2 & g_3 & g_4\\ -g_2 & g_1 & -g_4 & g_3\\ -g_3 & g_4 & g_1 & -g_2\\ -g_4 & -g_3 & g_2 & g_1 \end{bmatrix} \tag{15.4–41}$$

With $\{g_i\}$ equal to $\{s_i\}$ in the generator matrix in Equation 15.4–41, this code transmits four symbols in four consecutive time slots. Hence, $R_s = 1$ for this code.

Real orthogonal generator matrices are suitable for transmitting PAM signal constellations and square QAM signal constellations that can be decoupled into two separate PAM signal constellations. Real orthogonal generator matrix designs provide a diversity of order $N_T N_R$ and result in simple maximum-likelihood decoding by decoupling the decision for each transmitted symbol.

The orthogonality property which results in a low-complexity maximum-likelihood detector can be achieved for $N > 8$ at the cost of a lower spatial rate. Such space-time block codes are called generalized orthogonal codes and are defined by a $K \times N$ generator matrix $\boldsymbol{G}$ with real entries $g_1, -g_1, g_2, -g_2, \ldots, g_K, -g_K$, that satisfies the property

$$\boldsymbol{G}^t \boldsymbol{G} = b \left(\sum_{i=1}^{K} g_i^2 \right) \boldsymbol{I}_N$$

where b is a constant. The spatial rate is $R_s = K/N$.

The Alamouti code is an example of an orthogonal complex matrix design for $N_T = 2$. It has been shown in the literature (see Jafarkhani (2005) and Tarokh et al. (1999a)) that orthogonal complex matrix designs with $R_s = 1$ do not exist for $N_T > 2$ transmit antennas. However, by reducing the code rate, it is possible to devise complex orthogonal designs for two-dimensional signal constellations. For example, an orthogonal generator matrix for a STBC that transmits four complex-valued (PSK or QAM) symbols on $N_T = 4$ transmit antennas is

$$\boldsymbol{G} = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ -s_2 & s_1 & -s_4 & s_3 \\ -s_3 & s_4 & s_1 & -s_2 \\ -s_4 & -s_3 & s_2 & s_1 \\ s_1^* & s_2^* & s_3^* & s_4^* \\ -s_2^* & s_1^* & -s_4^* & s_3^* \\ -s_3^* & s_4^* & s_1^* & -s_2^* \\ -s_4^* & -s_3^* & s_2^* & s_1^* \end{bmatrix} \tag{15.4–42}$$

For this code generator, the four complex-valued symbols are transmitted in eight consecutive time slots. Hence the spatial rate for this code is $R_s = 1/2$. We also observe that

$$\boldsymbol{G}^H \boldsymbol{G} = \sum_{i=1}^{4} \left[|s_i|^2 \right] \boldsymbol{I}_4 \tag{15.4–43}$$

so that this code provides fourth-order diversity in the case of one receive antenna and $4N_R$ diversity with N_R receive antennas.

Complex orthogonal matrices with rate $R_s \leq 1/2$ exist for any number of transmit antennas. However, Wang and Xia (2003) have shown that complex orthogonal matrices for rates $R_s > 3/4$ do not exist. Rate $R_s = 3/4$ complex orthogonal matrices do exist. The following $R_s = 3/4$ complex orthogonal generator matrices are given in the paper by Tarokh et al. (1999a) for $N_T = 3$ and $N_T = 4$ transmit antennas:

$$G = \begin{bmatrix} s_1 & s_2 & s_3/\sqrt{2} \\ -s_2^* & s_1^* & s_3/\sqrt{2} \\ s_3^*/\sqrt{2} & s_3^*/\sqrt{2} & (-s_1 - s_1^* + s_2 - s_2^*)/2 \\ s_3^*/\sqrt{2} & -s_3^*/\sqrt{2} & (s_2 + s_2^* + s_1 - s_1^*)/2 \end{bmatrix} \tag{15.4–44}$$

$$G = \begin{bmatrix} s_1 & s_2 & s_3/\sqrt{2} & s_3/\sqrt{2} \\ -s_2^* & s_1^* & s_3/\sqrt{2} & -s_3/\sqrt{2} \\ s_3^*/\sqrt{2} & s_3^*/\sqrt{2} & (-s_1 - s_1^* + s_2 - s_2^*)/2 & (-s_2 - s_2^* + s_1 - s_1^*)/2 \\ s_3^*/\sqrt{2} & -s_3^*/\sqrt{2} & (s_2 + s_2^* + s_1 - s_1^*)/2 & -(s_1 + s_1^* + s_2 - s_2^*)/2 \end{bmatrix} \tag{15.4–45}$$

Finally, we should indicate that orthogonal generator matrix designs are not unique. To demonstrate this point, let U denote a unitary matrix, i.e., $U^H U = I$, and let G be a complex orthogonal matrix. Define $G_u = UG$. Then

$$\begin{aligned} G_u^H G_u &= (UG)^H UG \\ &= G^H U^H U G \\ &= G^H G \end{aligned} \tag{15.4–46}$$

Hence, a system employing the generator matrix G_u has the same properties as a system employing G.

Quasi-orthogonal Space-Time Block Codes As we have observed, orthogonal STBCs have the desirable property that the maximum-likelihood (ML) detector reduces to one that detects each symbol separately. Furthermore, for $N = 2, 4$, and 8, a real orthogonal STBC yields full diversity. Similarly, for $N = 2$, the Alamouti code with complex elements yields full diversity. We also observed that by reducing the code rate, it is possible to design (generalized) orthogonal codes having either real or complex elements. Thus, the low complexity of separate symbol detection can be maintained at the expense of a reduced rate and diversity.

On the other hand, we may relax the orthogonality condition which results in separate ML detection and attempt to design STBC with spatial rate $R_s = 1$ and full diversity. The simplest detector of such a design is one that allows for pairwise ML symbol detection. Such a code is called quasi-orthogonal. For example, a complex quasi-orthogonal STBC with rate $R_s = 1$ is specified by the generator matrix

$$G = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ -s_2^* & s_1^* & -s_4^* & s_3^* \\ -s_3^* & -s_4^* & s_1^* & s_2^* \\ s_4 & -s_3 & -s_2 & s_1 \end{bmatrix}$$

The transmitted symbols for this code can be optimally detected by a pairwise ML detector, and the code yields full diversity (see Problem 15.23).

Differential Space-Time Block Codes

In the application of the Alamouti code, as we have observed, it is assumed that the channel path coefficients $\{h_{ij}\}$ are constant over two successive time intervals. For $N_T > 2$ transmit antennas, the time interval over which the channel path coefficients are assumed to be constant is even larger. For example, the STBCs given in Equations 15.4–41, 15.4–44, and 15.4–45 are constructed based on the assumption that the channel path coefficients are constant over four time intervals. In a fading channel, this assumption is usually not satisfied precisely. That is, in practice, the channel path coefficients vary to some extent from one time interval to another. Consequently, the performance of the coherent detector may be degraded by the channel variation from one signal interval to the next. Further deterioration in the performance of the detector is caused by noisy estimates of the channel path coefficients $\{h_{ij}\}$. Typically, in practical systems, the transmitter sends pilot signals that the receiver uses to obtain estimates of the channel path coefficients. Then the estimates are used in the demodulation and detection of the STBC. In general, these estimates are noisy and cause some deterioration in the performance of the system. The effects of channel time variations and noisy channel estimates on the performance of the STBC have received considerable attention in the technical literature, e.g., Tarokh et al. (1999b), Buehrer and Kumar (2002), Gu and Leung (2003), and Jootar et al. (2005).

In rapidly fading channels, where the channel time variations preclude the use of coherent STBC, one may employ differential space-time modulation, which is akin to differential PSK (DPSK). Differential STBCs do not require knowledge of the channel path coefficients at the receiver. Consequently, the detector performs differentially coherent detection. As a result, the performance achieved by a differential STBC on a Rayleigh fading channel is approximately 3 dB worse than the performance of a coherently detected STBC. Differential STBCs are described in the papers by Tarokh and Jafarkhani (2000), Hughes (2000), Hochwald and Sweldens (2000), Tao and Cheng (2001), Jafarkhani and Tarokh (2001), Jafarkhani (2003), and Chen et al. (2003).

15.4–4 Pairwise Error Probability for a Space-Time Code

In this section we derive an expression for the pairwise error probability for a space-time coded MIMO system that is communicating over a frequency-nonselective Rayleigh fading channel. The MIMO system is assumed to employ a STBC for N_T transmit antennas and have spatial rate $R_s = N_T/N$, where N is the block length (number of time slots used to transmit the block code).

Let us denote the signal elements transmitted in each time slot by the vector $\boldsymbol{s}(l) = [s_1(l)\, s_2(l) \cdots s_{N_T}(l)]^t$ for $1 \le l \le N$ and let the space-time codeword be denoted by the $N_T \times N$ matrix $\boldsymbol{S} = [\boldsymbol{s}(1)\, \boldsymbol{s}(2) \cdots \boldsymbol{s}(N)]$. Then the transmitted signal may be expressed in matrix form as

$$\boldsymbol{X} = \sqrt{\frac{\mathcal{E}_s}{N_T}}\boldsymbol{S} \tag{15.4–47}$$

and the received signal may be expressed as

$$\boldsymbol{Y} = \sqrt{\frac{\mathcal{E}_s}{N_T}}\boldsymbol{H}\boldsymbol{S} + \boldsymbol{N} \tag{15.4–48}$$

where $\boldsymbol{H}$ is the $N_R \times N_T$ channel matrix with path coefficients $\{h_{ij}\}$, which are constant over the entire codeword, $\boldsymbol{Y} = [\boldsymbol{y}(1)\, \boldsymbol{y}(2) \cdots \boldsymbol{y}(N)]$ with

$$\boldsymbol{y}(l) = \sqrt{\frac{\mathcal{E}_s}{N_T}} \boldsymbol{H}\boldsymbol{s}(l) + \boldsymbol{\eta}(l), \qquad 1 \le l \le N \tag{15.4–49}$$

and $\boldsymbol{N} = [\boldsymbol{\eta}(1)\, \boldsymbol{\eta}(2) \ldots \boldsymbol{\eta}(N)]$ represents the additive noise. The noise components are assumed to be statistically independent and identically distributed, zero-mean, complex-valued Gaussian with variance N_0.

The receiver employs a maximum-likelihood (ML) decoder that is assumed to know the channel matrix $\boldsymbol{H}$. Since the additive noise components are iid, the decoder searches for the valid codeword that is closest in Euclidean distance to the received codeword. Thus, the decoder output is

$$\tilde{\boldsymbol{S}} = \arg \min_{\underline{\boldsymbol{S}}} \|\boldsymbol{Y} - \boldsymbol{H}\boldsymbol{S}\|_F^2 \tag{15.4–50}$$

Let us assume that the codeword $\boldsymbol{S}^{(k)}$ was transmitted. Then the pairwise error probability (PEP) that $\boldsymbol{S}^{(j)}$ is selected when $\boldsymbol{S}^{(k)}$ is transmitted, for any given channel matrix realization, is

$$P(\boldsymbol{S}^{(k)} \to \boldsymbol{S}^{(j)} | \boldsymbol{H}) = Q\left(\sqrt{\frac{\mathcal{E}_s}{2N_0N_T} \, \|\boldsymbol{H}(\boldsymbol{S}^{(k)} - \boldsymbol{S}^{(j)})\|_F^2}\right) \tag{15.4–51}$$

It is convenient to define an $N_T \times N$ error matrix as $\boldsymbol{E}_{kj} = \boldsymbol{S}^{(k)} - \boldsymbol{S}^{(j)}$ and to approximate the PEP by the Chernov bound

$$P(\boldsymbol{S}^{(k)} \to \boldsymbol{S}^{(j)} | \boldsymbol{H}) \le \exp\left\{\frac{-\mathcal{E}_s}{4N_0N_T} \, \|\boldsymbol{H}\boldsymbol{E}_{k_j}\|_F^2\right\} \tag{15.4–52}$$

We can now average this conditional PEP over the statistics of the channel matrix $\boldsymbol{H}$. Assuming that the channel path coefficients $\{h_{ij}\}$ are iid, complex-valued zero-mean Gaussian (spatially white channel), the average of the PEP in Equation 15.4–52 over the statistics of the channel path coefficients yields the upper bound on the average PEP as

$$\begin{aligned} P(\boldsymbol{S}^{(k)} \to \boldsymbol{S}^{(j)}) &\le \frac{1}{\left[\det\left(\boldsymbol{I}_{N_T} + \dfrac{\mathcal{E}_s}{4N_0N_T}\boldsymbol{E}_{kj}\boldsymbol{E}_{kj}^H\right)\right]^{N_R}} \\ &\le \left(\prod_{n=1}^{r} \frac{1}{1 + \dfrac{\mathcal{E}_s\lambda_n}{4N_0N_T}}\right)^{N_R} \end{aligned} \tag{15.4–53}$$

where r is the rank of the $N_T \times N_T$ matrix $\boldsymbol{A}_{kj} = \boldsymbol{E}_{kj}\boldsymbol{E}_{kj}^H$ and $\{\lambda_n\}$ are the nonzero eigenvalues of $\boldsymbol{A}_{kj}$.

At high SNR, where $\mathcal{E}_s/4N_0N_T \gg 1$, the bound on the PEP may be expressed as

$$P(S^{(k)} \to S^{(j)}) \le \left(\prod_{n=1}^{r} \lambda_n\right)^{-N_R} (\mathcal{E}_s/4N_0N_T)^{-rN_R} \tag{15.4–54}$$

This expression for the PEP suggests the following two criteria for designing ST codes, namely, the *rank criterion* and the *determinant criterion*, as described in the paper by Tarokh et al. (1998). In applying the rank criterion, the objective is to achieve the maximum possible diversity of $N_T N_R$, which is obtained when the matrix A_{kj} is full rank ($r = N_T$) for any pair of valid codewords. If A_{kj} has minimum rank r for a pair of codewords, the order of diversity is rN_R. In applying the determinant criterion, the objective is to maximize the minimum of the determinant of matrix A_{kj} taken over all pairs(k, j) of valid codewords. The term in the PEP involving the product of the nonzero eigenvalues of A_{kj} has been coined as the coding gain of the space-time code. Hence, the determinant criterion has the objective of maximizing the coding gain of the space-time code.

15.4–5 Space-Time Trellis Codes for MIMO Channels

We observed in Section 8.12 that trellis-coded modulation (TCM) is a combination of a trellis code and an appropriately selected signal constellation designed with the aim of achieving a coding gain. Space-time trellis coding also combines trellis coding and a selected signal constellation with the primary objective of achieving the maximum possible spatial diversity at the highest code rate. To achieve this objective, code construction may be based on applying the rank criterion and the determinant criterion described in Section 15.4–4.

In applying the rank criterion, we optimize the spatial diversity obtained from the space-time code, or equivalently we maximize the rank of the matrices $A_{ij} = (S^{(i)} - S^{(j)})(S^{(i)} - S^{(j)})^H$ over all pairs (i, j) of codewords. The goal is to achieve the full rank of N_T. It has been shown (see Jafarkhani (2005)) that for a bit rate of b bps/Hz and a diversity r, a space-time trellis code (STTC) must have at least $2^{b(r-1)}$ states. Thus, to achieve full diversity, a STTC must have at least $2^{b(N_T-1)}$ states.

Space-time trellis codes may be designed either manually or with the aid of a computer by following some simple rules, similar in nature to the rules formulated by Ungerboeck (1982) for designing trellis codes for TCM. Tarokh et al. (1998) specify two design rules that guarantee full diversity for MIMO systems with two transmit antennas.

Design Rule 1: Transitions departing from the same state should differ in the second symbol (symbol transmitted on the second antenna).

Design Rule 2: Transitions arriving at the same state should differ in the first symbol (symbol transmitted on the first antenna).

As an example of a STTC, we consider the 4-state trellis code shown in Figure 15.4–6, which is designed for two transmit antennas and QPSK modulation.

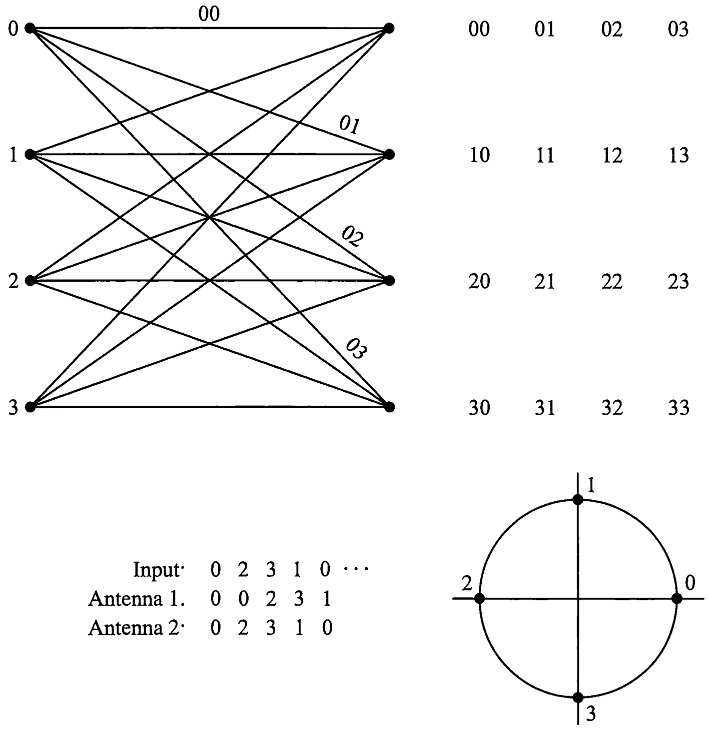


FIGURE 15.4–6
4-PSK, 4-state, space-time trellis code.

The states are denoted as $S_t = 0, 1, 2, 3$. The input to the encoder is a pair of bits (00, 01, 10, 11) which are mapped into the corresponding phases that are numbered (0, 1, 2, 3), respectively. The indices 0, 1, 2, 3 correspond to the four phases, which are called symbols. Initially, the encoder is in state $S_t = 0$. Then for each pair of input bits, which are mapped into a corresponding symbol, the encoder generates a pair of symbols, the first of which is transmitted on the first antenna, and the second symbol is transmitted simultaneously on the second antenna. For example, when the encoder is in state $S_t = 0$ and the input bits are 11, the symbol is a 3. The STTC outputs the pair of symbols (0, 3), corresponding to the phases 0 and $3\pi/2$. The zero phase signal is transmitted in the first antenna, and the $3\pi/2$ phase signal is transmitted on the second antenna. At this point the encoder goes to state $S_t = 3$. If the next two input bits are 01, the encoder outputs the symbols (3, 1) which are transmitted on the two antennas. Then, the encoder goes to state $S_t = 1$, and this procedure continues. At the end of a block of input bits, say a frame of data, zeros are inserted in the data stream to return the encoder to the state $S_t = 0$. Thus the STTC transmits at a bit rate of 2 bps/Hz. We note that it satisfies the two design rules given above and achieves full rank of $N_T = 2$.

Increasing the number of states in the trellis beyond 2^b states allows the designer to increase the coding gain by increasing the product of the eigenvalues (determinant) in the expression for the pairwise error probability. For example, the 8-state STTC, given in the paper by Tarokh et al. (1998), that transmits at a bit rate of 2 bps/Hz with QPSK modulation is shown in Figure 15.4–7. This code provides the same diversity order ($2N_R$) as the 4-state STTC illustrated in Figure 15.4–6, but achieves a larger coding gain.

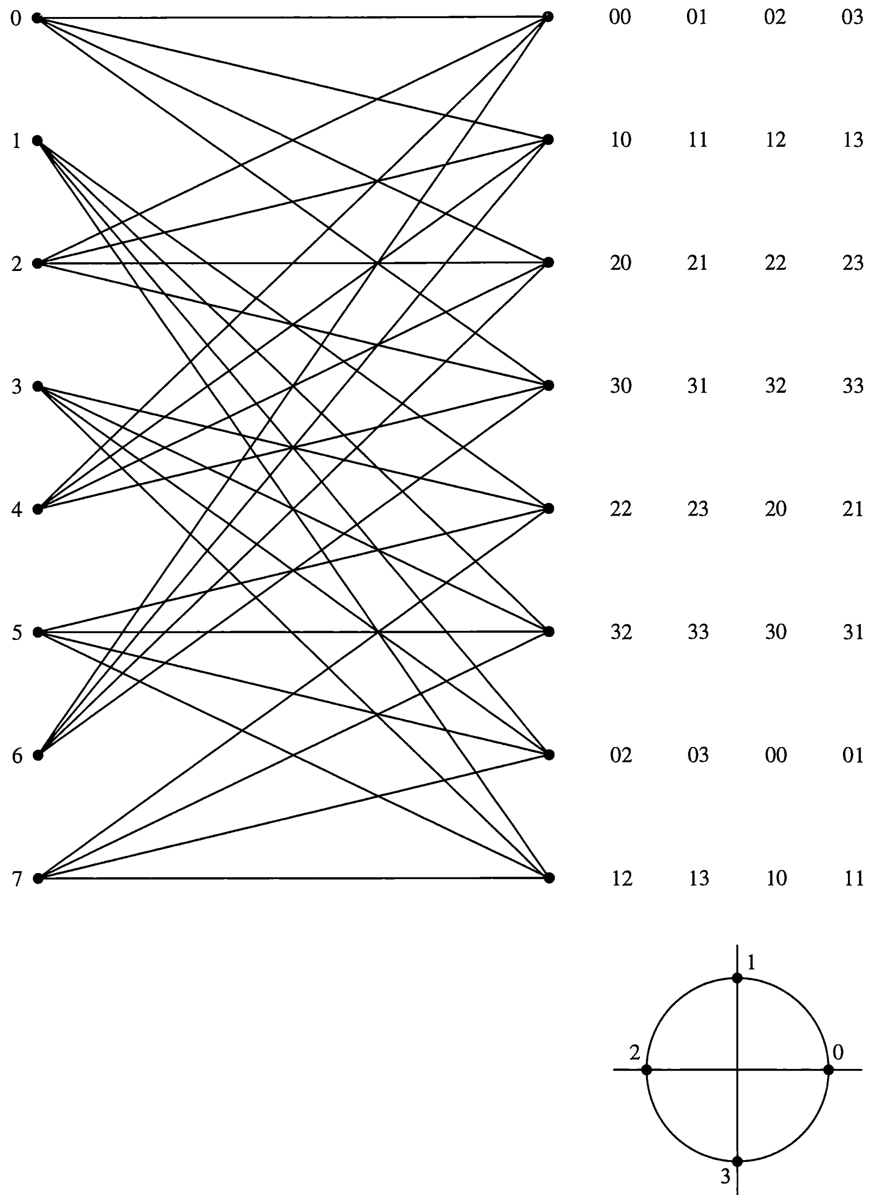

0
1
2
3
4
5
6
7
00 01 02 03
10 11 12 13
20 21 22 23
30 31 32 33
22 23 20 21
32 33 30 31
02 03 00 01
12 13 10 11
1
2
0
3

FIGURE 15.4–7
4-PSK, 8-state, space-time trellis code.

The paper by Tarokh et al. (1998) also describes higher rate codes for two transmit antennas. For example, Figure 15.4–8 illustrates an 8-state STTC for use with 8-PSK modulation to achieve a bit rate of 3 bps/Hz and full diversity of $N_T = 2$. STTC for large constellations employing QAM are given in the paper by Tarokh et al. (1998) and other publications in the literature.

In decoding a STTC, the maximum-likelihood sequence detection (MLSD) criterion provides the optimum performance. MLSD is efficiently implemented by use of the

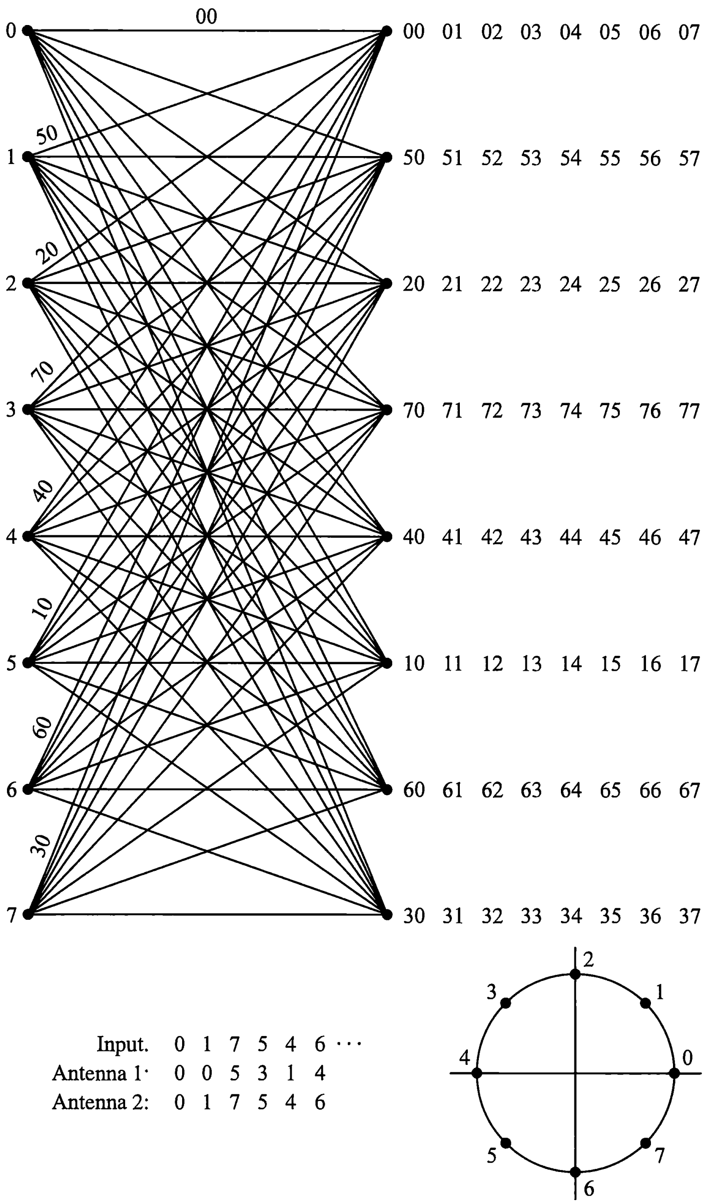

0
1
2
3
4
5
6
7
00
50
20
70
40
10
60
30
00 01 02 03 04 05 06 07
50 51 52 53 54 55 56 57
20 21 22 23 24 25 26 27
70 71 72 73 74 75 76 77
40 41 42 43 44 45 46 47
10 11 12 13 14 15 16 17
60 61 62 63 64 65 66 67
30 31 32 33 34 35 36 37
Input. 0 1 7 5 4 6 ···
Antenna 1· 0 0 5 3 1 4
Antenna 2: 0 1 7 5 4 6
2
3
1
4
0
5
7
6

FIGURE 15.4–8
8-PSK, 8-state, space-time trellis code.

Viterbi algorithm. For two transmit antennas., the branch metrics may be expressed as

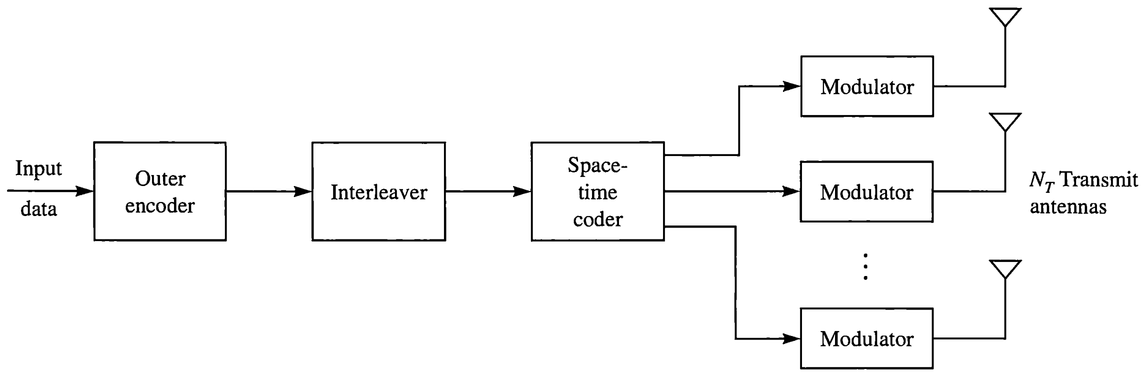
$$\mu_b(s_1, s_2) = \sum_{j=1}^{N_R} |y_j - h_{1j}\, s_1 - h_{2j}\, s_2|^2 \tag{15.4–55}$$

where $\{y_j, 1 \leq j \leq N_R\}$ are the outputs of the matched filters at the N_R receive antennas, $\{h_{1j}, 1 \leq j \leq N_R\}$ and $\{h_{2j}, 1 \leq j \leq N_R\}$ are the channel coefficients in a frequency-nonselective channel, and (s_1, s_2) denote the symbols transmitted on the two antennas. By using these branch metrics in the Viterbi algorithm, to form the path metrics of valid paths through the trellis, we can find the path that minimizes the overall metric and thus determine the sequence of transmitted symbols corresponding to the path having the smallest path metric.

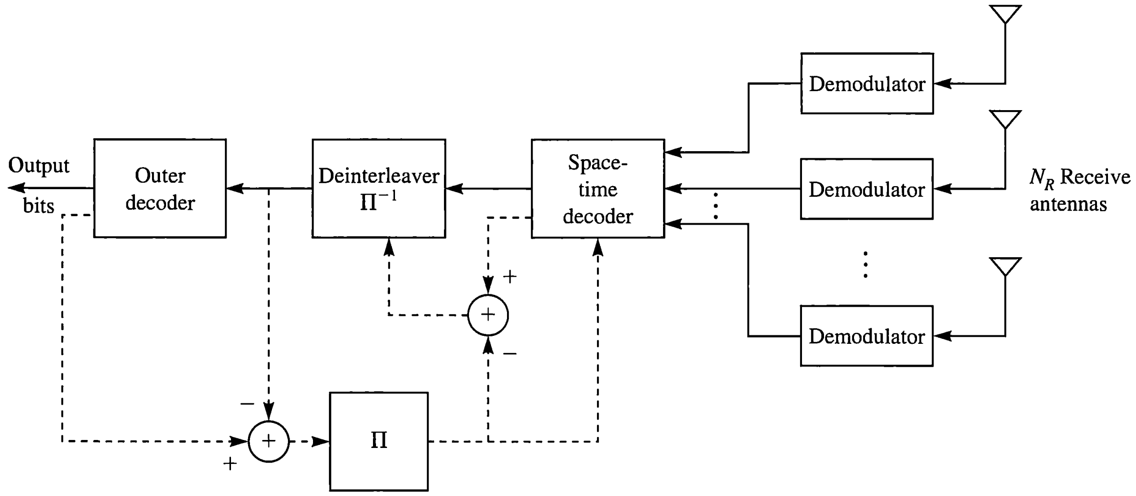
15.4–6 Concatenated Space-Time Codes and Turbo Codes

In Section 15.4–2, we observed that temporal coding with interleaving provides a means to achieve diversity in a MIMO system. It is also possible to construct concatenated codes using temporal coding with interleaving in combination with space-time codes. Figure 15.4–9 illustrates a system in which the input data stream is temporally coded by either a block code or a convolutional code. Following the temporal encoding, the data are bit-interleaved and passed to the space-time encoder, which may be either a STBC or a STTC.

At the receiver, the space-time code is decoded first, and its output is deinterleaved and passed to the outer decoder. The output of the outer decoder constitutes the


Input
data
Outer
encoder
Interleaver
Space-
time
coder
Modulator
Modulator
Modulator
N_T Transmit
antennas

(a) Transmitter


Output
bits
Outer
decoder
Deinterleaver
Π^{-1}
Space-
time
decoder
Demodulator
Demodulator
Demodulator
N_R Receive
antennas
+
−
−
+
Π

(b) Receiver

FIGURE 15.4–9
A MIMO system with concatenated coding consisting of a temporal outer code and a space-time inner code (dotted lines in the receiver indicate iterative decoding).

reconstructed data stream. If desired, iterative decoding can be performed between the inner and outer decoders by making multiple passes on the received data signal. Such iterative decoding leads to an improvement in system performance but at a significant cost in implementation (computational) complexity.

A turbo code (parallel concatenated convolutional encoders separated by an interleaver) can also be used as the outer code in a concatenated coding scheme, as shown in Figure 15.4–9. In such a case, the outer decoder at the receiver is a turbo (iterative) decoder. Iterative decoding can also be implemented between the turbo decoder and the space-time decoder. However, iterative decoding between the inner space-time decoder and the turbo decoder significantly increases the computational complexity of the receiver.

15.5

BIBLIOGRAPHICAL NOTES AND REFERENCES

The use of multiple antennas at the receiver of the communication system has been a well-known method for achieving spatial diversity to combat fading without expanding the bandwidth of the transmitted signal. Much less common has been the use of multiple antennas at the transmitter to achieve spatial diversity. The publications of Wittneben (1993) and Seshadri and Winters (1994) are two of the early publications on this topic.

A major breakthrough occurred with the publications of Foschini (1996) and Foschini and Gans (1998), which demonstrated that multiple antennas at the transmitter and the receiver of a wireless communication system can be used to establish multiple parallel channels for simultaneous transmission of multiple data streams in the same frequency band (spatial multiplexing) and, thus, result in extremely high bandwidth efficiency. Since then, there have been numerous publications on the analysis of the performance characteristics of MIMO wireless communication systems and their implementation in practical systems. Basic treatments of MIMO systems may be found in the textbooks by Goldsmith (2005), Haykin and Moher (2005), and Tse and Viswanath (2005).

Pioneering work on space-time coding for MIMO channels was performed by Tarokh et al. (1998, 1999a). The book by Jafarkhani (2005) provides a comprehensive treatment of both space-time block codes and trellis codes.

PROBLEMS

15.1 Consider an $(N_T, N_R) = (2, 1)$ MIMO system that employs the Alamouti code to transmit a binary sequence using binary PSK modulation. The channel is Rayleigh fading characterized by the channel vector

$$\boldsymbol{h} = [h_{11}\ h_{12}]^t$$

with $E|h_{11}|^2 = E|h_{12}|^2 = 1$. The additive noise is zero-mean Gaussian. Determine the average probability of error for the system.

15.2 Consider a SIMO AWGN channel with N_R receive antennas. Instead of maximal ratio combining, the receiver selects the signal from the antenna having the strongest signal; i.e., if $\boldsymbol{h} = [h_1, h_2, \ldots, h_{N_R}]$ is the channel vector, the receiver selects the antenna with channel coefficient

$$|h_{\max}| = \max |h_i|, \qquad i = 1, 2, \ldots, N_R$$

This method is called selection diversity. Determine the capacity of a MIMO system that employs selection diversity.

15.3 Prove the relationship between the eigenvalues of $\boldsymbol{H}\boldsymbol{H}^H$ and the singular values of the channel matrix $\boldsymbol{H}$, as given by Equation 15.2–4.

15.4 Consider a MIMO system with $N_R = N_T = N$ antennas and AWGN. The ergodic capacity for the MIMO system is

$$\bar{C} = E\left[\sum_{i=1}^{N} \log_2\left(1 + \frac{\mathcal{E}_s}{NN_0}\lambda i\right)\right]$$

Show that for N large, the capacity can be approximated as

$$C \approx \frac{\mathcal{E}_s}{N_0 \ln 2}\lambda_{\text{av}}$$

where λ_{av} is the average of the eigenvalues of $\boldsymbol{H}\boldsymbol{H}^H$.

15.5 Consider a deterministic SIMO channel with AWGN in which the elements of the channel vector $\boldsymbol{h}$ satisfy the conditions $|h_i|^2 = 1$, $i = 1, 2, \ldots, N_R$.

a. Determine the capacity of this SIMO channel when $\boldsymbol{h}$ is known at the receiver only.

b. Suppose that $\boldsymbol{h}$ is also known at the transmitter. Does this additional knowledge increase the channel capacity? Explain.

15.6 Consider a deterministic MISO channel with AWGN in which the elements of the channel vector $\boldsymbol{h}$ satisfy the conditions $|h_i|^2 = 1$, $i = 1, 2, \ldots, N_T$.

a. Determine the capacity of this MISO channel when $\boldsymbol{h}$ is known at the receiver only.

b. How does this capacity compare with that of a SIMO and a SISO channel?

15.7 Consider a MIMO system with $N_R = N_T = N$ antennas and AWGN. The rank of the channel matrix $\boldsymbol{H}$ is N.

a. Show that the capacity

$$C = \sum_{i=1}^{N} \log_2\left(1 + \frac{E_s}{NN_0}\lambda_i\right)$$

subject to the constraint that

$$\sum_{i=1}^{N} \lambda_i = \beta = \text{constant}$$

is maximized when $\lambda_i = \beta/N$ for $i = 1, 2, \ldots, N$, and hence

$$C = N \log_2 \left(1 + \frac{\beta \mathcal{E}_s}{N^2 N_0}\right)$$

b. If $\lambda_i = \beta/N$ for $i = 1, 2, \ldots, N$, show that $\boldsymbol{H}$ must be an orthogonal matrix that satisfies the condition

$$\boldsymbol{H}\boldsymbol{H}^H = \boldsymbol{H}^H\boldsymbol{H} = \frac{\beta}{N}\boldsymbol{I}_N$$

c. Show that if all the elements of $\boldsymbol{H}$ are unit magnitude, i.e., $|H_{ij}| = 1$, then $\|H\|_F^2 = N^2$ and

$$C = N \log_2 \left(1 + \frac{\mathcal{E}_s}{N_0}\right)$$

Hence, under these conditions, the capacity of the orthogonal MIMO channel is N times the capacity of a SISO channel.

15.8 The received signal vector in a frequency-nonselective AWGN MIMO channel with N_T transmit antennas and N_R receive antennas is given by Equation 15.2–7 as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{\eta}$$

a. Use the SVD to transform the received signal vector to the form

$$\boldsymbol{y}' = \boldsymbol{\Sigma}\boldsymbol{s} + \boldsymbol{\eta}'$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix of rank r with the nonzero diagonal elements equal to the singular values of the channel matrix $\boldsymbol{H}$.

b. Show that if the elements of $\boldsymbol{\eta}$ are statistically iid, zero-mean, complex-valued Gaussian random variables, then the elements of $\boldsymbol{\eta}'$ are also iid zero-mean complex-valued Gaussian random variables.

c. Show that the capacity of the AWGN MIMO channel may be expressed as

$$C = \sum_{k=1}^{r} \log_2 \left(1 + \frac{P_k \sigma_k^2}{N_0}\right) \qquad \text{bps/Hz}$$

where $P_1, P_2, \ldots, P_r$ are the allocated powers based on the water-filling criterion with the total power constraint

$$\sum_{k=1}^{r} P_k = P$$

15.9 The capacity of MISO channel with AWGN, when the channel is known at the receiver only, may be expressed as

$$C = \log_2 \left(1 + \frac{\gamma}{N_T} \sum_{i=1}^{N_T} |h_i|^2\right)$$

where γ is the SNR and $\boldsymbol{h} = [h_1\, h_2 \cdots h_{N_T}]^t$ is the channel coefficient vector. Suppose the channel coefficients are iid zero-mean, complex Gaussian distributed with $E|h_i|^2 = 1,\ i = 1, 2, \dots, N_T$.

a. Determine the PDF of the random variable

$$X = \sum_{i=1}^{N_T} |h_i|^2$$

b. Note that C is a monotonic function of X. Show that the outage probability for the MISO system may be expressed as

$$P_{\text{out}} = P\left[X \le N_T \frac{2^C - 1}{\gamma}\right]$$

c. Evaluate and plot P_{out} versus γ for $C = 2$ bps/Hz and $N_T = 1, 2, 4, 8$.

d. For $\gamma = 10$ dB, evaluate and plot the complementary cumulative distribution function (CCDF)

$$1 - P_{\text{out}} = P\left[X \ge N_T \frac{2^C - 1}{\gamma}\right]$$

versus C for $N = 1, 2, 4, 8$. This is the CCDF for the outage capacity. Repeat the computation for $\gamma = 20$ dB.

e. Let $P_{\text{out}} = 0.1$ (corresponding to 10% outage capacity) and plot C versus γ for $N_T = 1, 2, 4, 8$.

15.10 Consider a deterministic MISO $(N_T, 1)$ channel with AWGN and channel vector $\boldsymbol{h}$. The received signal in any signal interval may be expressed as

$$y = \boldsymbol{h}\boldsymbol{s} + \eta$$

where y and η are scalars.

a. If the channel vector $\boldsymbol{h}$ is known at the transmitter, demonstrate that the received SNR is maximized when the information is sent in the direction of the channel vector $\boldsymbol{h}$, i.e., $\boldsymbol{s}$ is selected as

$$\boldsymbol{s} = \frac{\boldsymbol{h}^*}{\|\boldsymbol{h}\|} s'$$

(The alignment of the transmit signal in the direction of the channel vector $\boldsymbol{h}$ is called transmit beamforming.)

b. What is the capacity of the MISO channel when $\boldsymbol{h}$ is known at the transmitter?

c. Compare the capacity obtained in (*b*) with that of a SIMO channel, when the channel matrix $\boldsymbol{h}$ is identical for the two systems.

15.11 Determine the outage probability of an $(N_T, N_R) = (4, 1)$ MIMO system for an SNR $\gamma = 20$ dB and outage capacity $C_{\text{out}} = 2$ bps/Hz.

15.12 The capacity of a SIMO channel with AWGN may be expressed as

$$C = \log_2\left(1 + \gamma \sum_{i=1}^{N_R} |h_i|^2\right)$$

where γ is the SNR and $\boldsymbol{h} = [h_1 \; h_2 \cdots h_{N_R}]^t$ is the channel coefficient vector. The channel coefficients are complex-valued, iid zero-mean Gaussian distributed with $E|h_i|^2 = 1$, $i = 1, 2, \ldots, N_R$.

a. Determine the PDF of the random variable

$$X = \sum_{i=1}^{N_R} |h_i|^2$$

b. Note that C is a monotonic function of X. Show that the outage probability for the SIMO system may be expressed as

$$P_{\text{out}} = P\left[X \leq \frac{2^C - 1}{\gamma}\right]$$

c. Evaluate and plot p_{out} versus γ for $C = 2$ bps/Hz and $N_R = 1, 2, 4, 8$.

d. For $\gamma = 10$ dB, evaluate and plot the complementary cumulative distribution function (CCDF)

$$1 - P_{\text{out}} = P\left[X \geq \frac{2^C - 1}{\gamma}\right]$$

versus C for $N_R = 1, 2, 4, 8$. This is the CCDF for the outage capacity. Repeat for $\gamma = 20$ dB.

e. Let $P_{\text{out}} = 0.1$ (corresponding to 10% outage capacity) and plot C versus γ for $N_R = 1, 2, 4, 8$.

15.13 Consider an $(N_T, N_R) = (2, N_R)$ MIMO system that employs the Alamouti code with QPSK modulation. If the input bit stream is 01101001110010, determine the transmitted symbols from each antenna for each signaling interval.

15.14 Show that the detector that computes the estimates $\hat{s}_1$ and $\hat{s}_2$ given by Equation 15.4–25 is equivalent to the detector that computes the correlation metrics in Equation 15.4–22.

15.15 Determine the decision variables for the separate ML decoding of the symbols in the following rate 3/4 block code.

$$\boldsymbol{C} = \begin{bmatrix} s_1 & s_2 & s_3 \\ -s_2^* & s_1^* & 0 \\ s_3^* & 0 & -s_1^* \\ 0 & s_3^* & -s_2^* \end{bmatrix}$$

15.16 Determine the decision variables for the separate ML decoding of the symbols in the rate 1/2 orthogonal STBC given by Equation 15.4–42.

15.17 Determine the probability of error for the detector with input metrics given by Equation 15.3–5 for BPSK modulation and a Rayleigh fading channel. Assume that the components of $\boldsymbol{h}_j$ are iid, zero-mean, complex-valued Gaussian random variables.

15.18 For a Rayleigh fading channel and BPSK modulation, determine the performance of a MISO (2, 1) system employing the Alamouti code with that of a SIMO (1, 2) system. Assume that the transmitter power is the same for the two systems.

15.19 Consider a MISO (2, 1) system in which the Alamouti code is used in conjunction with multicode spread spectrum. To be specific, suppose that the symbol s_1 is spread by code $\boldsymbol{c}_1$ and $-s_2^*$ is spread by code $\boldsymbol{c}_2$. These two spread spectrum signals are added and transmitted on antenna 1. Similarly, the symbol s_2 is spread by $\boldsymbol{c}_1$ and the symbol s_1^* is spread by the code $\boldsymbol{c}_2$. Then two spread spectrum signals are added and transmitted on antenna 2. The channel coefficients h_1 and h_2 are known at the receiver.

a. Sketch the block diagram configuration of the transmitter and the receiver, illustrating the modulation and demodulation operations.

b. Assuming that the spreading codes $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ are orthogonal, determine the expressions for the decision variables $\hat{s}_1$ and $\hat{s}_2$.

c. What, if any, are the advantages and disadvantages of this multicode MISO (2, 1) system over the conventional MISO (2, 1) system that employs the Alamouti STBC without the multicode spreading?

15.20 Consider an uncoded MIMO system with $N_T = N_R$ antennas that transmits over a frequency-nonselective channel in which the channel matrix $\boldsymbol{H}$ has iid complex-valued, zero-mean Gaussian elements. The received signal vector is

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{\eta}$$

where the elements of $\boldsymbol{\eta}$ are iid complex-valued, zero-mean Gaussian. The detector used at the receiver is the inverse channel detector (ICD), described in Section 15.1–2.

a. Determine the covariance matrix of the noise at the output of the detector.

b. If the detector makes independent decisions on each of the N_T transmitted symbols, is this detector optimum (in the sense of minimizing the error probability)?

c. If BPSK modulation is employed, determine the error probability of the detector described in (b).

d. Now, suppose that $N_R > N_T$ and the decisions made by the detector are based on the signal estimate $\hat{\boldsymbol{s}} = \boldsymbol{W}^H\boldsymbol{y}$, where $\boldsymbol{W}^H = (\boldsymbol{H}^H\boldsymbol{H})^{-1}\boldsymbol{H}^H$. Repeat parts (a) and (b).

15.21 The channel matrix in an $N_T = N_R = 2$ MIMO system with AWGN is

$$\boldsymbol{H} = \begin{bmatrix} 0.4 & 0.5 \\ 0.7 & 0.3 \end{bmatrix}$$

a. Determine the SVD of $\boldsymbol{H}$.

b. Based on the SVD of $\boldsymbol{H}$, determine an equivalent MIMO system having two independent channels, and find the optimal power allocation and channel capacity when $\boldsymbol{H}$ is known at the transmitter and the receiver.

c. Determine the channel capacity when $\boldsymbol{H}$ is known only at the receiver.

15.22 Consider the following two MISO (2, 1) systems with AWGN. The first employs the Alamouti code to achieve transmit diversity when the channel is known only at the receiver. The second MISO (2, 1) also achieves transmit diversity, but the channel is known at the transmitter. Determine and compare the outage probabilities for the two systems. Which MISO system has a lower outage probability for the same SNR?

15.23 The generator matrix for a rate $R_s = 1$ STBC is given as

$$\boldsymbol{G} = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ -s_2^* & s_1^* & -s_4^* & s_3^* \\ -s_3^* & -s_4^* & s_1^* & s_2^* \\ s_4 & -s_3 & -s_2 & s_1 \end{bmatrix}$$

a. Determine the matrix $\boldsymbol{G}^H \boldsymbol{G}$, and thus show that the code is not orthogonal.
b. Show that the ML detector can perform pairwise ML detection.
c. What is the order of diversity achieved by this code?

16

Multiuser Communications

In the MIMO communication systems that were treated in Chapter 15, we observed that multiple data streams can be sent simultaneously from a transmitter employing multiple antennas to a receiver that employs multiple receive antennas. This type of a MIMO system is generally viewed as a single-user point-to-point communication system, having the primary objectives of increasing the data rate through spatial multiplexing and improving the error rate performance by increasing signal diversity to combat fading. In this chapter, the focus shifts to multiple users and multiple communication links. We explore the various ways in which multiple users access a common channel to transmit information. The multiple access methods that are described in this chapter form the basis for current and future wireline and wireless communication networks, such as satellite networks, cellular and mobile communication networks, and underwater acoustic networks.

16.1
INTRODUCTION TO MULTIPLE ACCESS TECHNIQUES

It is instructive to distinguish among several types of multiuser communication systems. One type is a multiple access system in which a large number of users share a common communication channel to transmit information to a receiver. A model of such a system is depicted in Figure 16.1–1. The common channel may represent the uplink in either a cellular or a satellite communication system, or a cable to which are connected a number of terminals that access a central computer. For example, in a mobile cellular communication system, the users are the mobile terminals in any particular cell of the system, and the receiver resides in the base station of the particular cell.

A second type of multiuser communication system is a broadcast network in which a single transmitter sends information to multiple receivers, as depicted in Figure 16.1–2. Examples of broadcast systems include the common radio and TV broadcast systems as well as the downlinks in cellular and satellite communication systems.

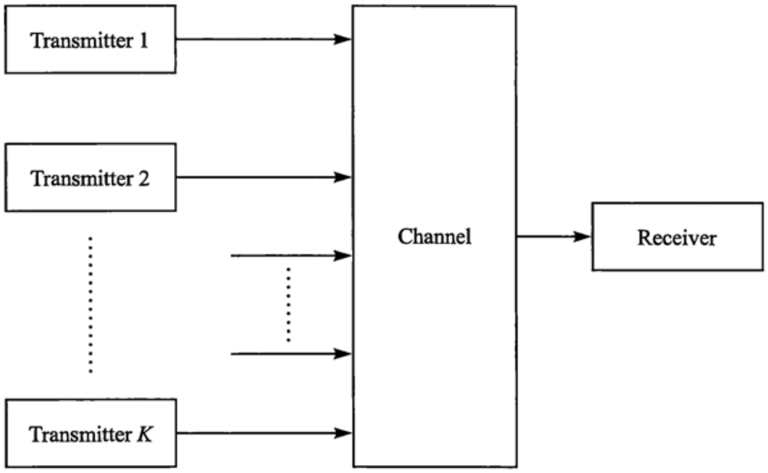


FIGURE 16.1–1
A multiple access system.

The multiple access and broadcast systems are the most common multiuser communication systems. A third type of multiuser system is a store-and-forward network, as depicted in Figure 16.1–3. Yet a fourth type is the two-way communication system shown in Figure 16.1–4.

In this chapter, we focus on multiple access and broadcast methods for multiuser communications. In a multiple access system, there are several different ways in which multiple users can send information through the communication channel to the receiver. One simple method is to subdivide the available channel bandwidth into a number, say K, of frequency non-overlapping subchannels, as shown in Figure 16.1–5, and to assign a subchannel to each user upon request by the users. This method is generally called *frequency-division multiple access* (FDMA) and is commonly used in wireline channels to accommodate multiple users for voice and data transmission.

Another method for creating multiple subchannels for multiple access is to subdivide the duration T_f, called the *frame duration*, into, say, K non-overlapping subintervals, each of duration T_f/K. Then each user who wishes to transmit information

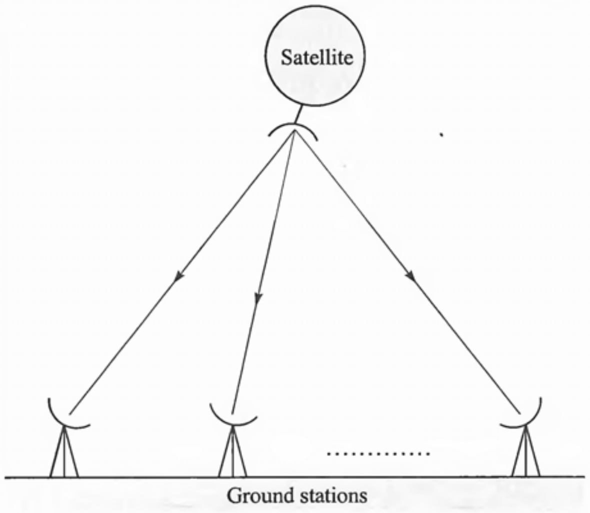


FIGURE 16.1–2
A broadcast network.

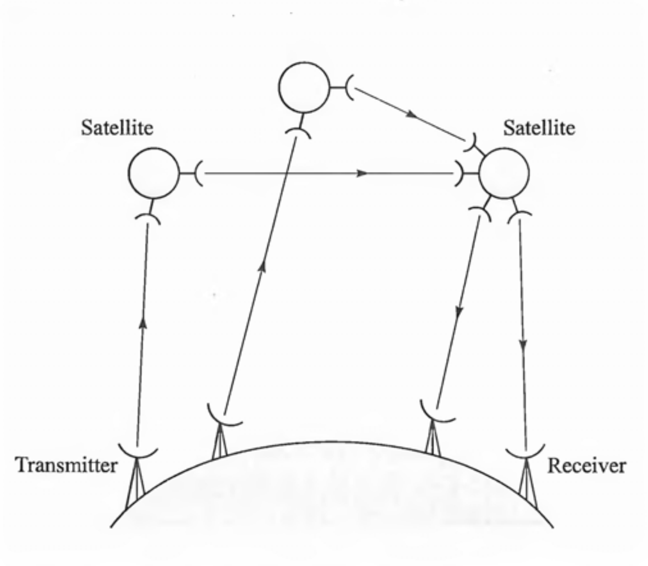


FIGURE 16.1–3
A store-and-forward communication network with satellite relays.

is assigned to a particular time slot within each frame. This multiple access method is called *time-division multiple access* (TDMA) and it is frequently used in data and digital voice transmission.

We observe that in FDMA and TDMA, the channel is basically partitioned into independent single-user subchannels. In this sense, the communication system design methods that we have described for single-user communication are directly applicable and no new problems are encountered in a multiple access environment, except for the additional task of assigning users to available channels.

The interesting problems arise when the data from the users accessing the network is bursty in nature. In other words, the information transmissions from a single user are separated by periods of no transmission, where these periods of silence may be greater than the periods of transmission. Such is the case generally with users at various terminals in a computer communication network. To some extent, this is also the case in mobile cellular communication systems carrying digitized voice, since speech signals typically contain long pauses.

In such an environment where the transmission from the various users is bursty and low-duty-cycle, FDMA and TDMA tend to be inefficient because a certain percentage of the available frequency slots or time slots assigned to users do not carry information. Ultimately, an inefficiently designed multiple access system limits the number of simultaneous users of the channel.

An alternative to FDMA and TDMA is to allow more than one user to share a channel or subchannel by use of direct-sequence spread spectrum signals. In this

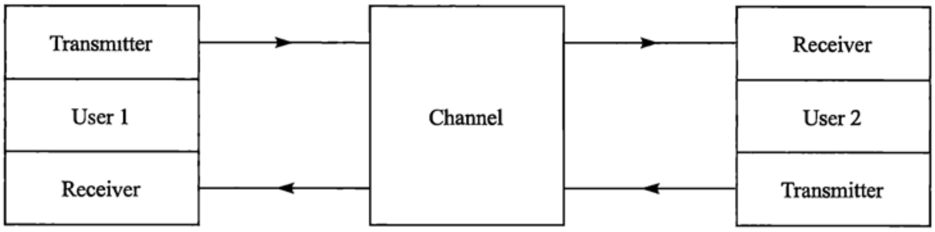


FIGURE 16.1–4
A two-way communication channel.

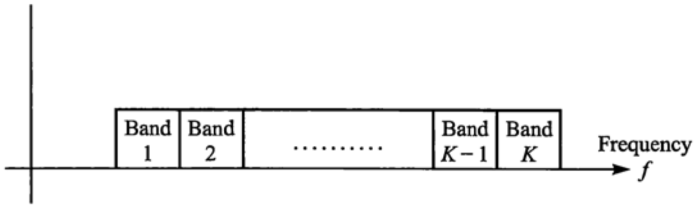


FIGURE 16.1–5
Subdivision of the channel into non-overlapping frequency bands.

method, each user is assigned a unique code sequence or *signature sequence* that allows the user to spread the information signal across the assigned frequency band. Thus signals from the various users are separated at the receiver by cross correlation of the received signal with each of the possible user signature sequences. By designing these code sequences to have relatively small cross-correlations, the crosstalk inherent in the demodulation of the signals received from multiple transmitters is minimized. This multiple access method is called *code division multiple access* (CDMA).

In CDMA, the users access the channel in a random manner. Hence, the signal transmissions among the multiple users completely overlap both in time and in frequency. The demodulation and separation of these signals at the receiver is facilitated by the fact that each signal is spread in frequency by the pseudorandom code sequence. CDMA is sometimes called *spread spectrum multiple access* (SSMA).

An alternative to CDMA is nonspread random access. In such a case, when two users attempt to use the common channel simultaneously, their transmissions collide and interefere with each other. When that happens, the information is lost and must be retransmitted. To handle collisions, one must establish protocols for retransmission of messages that have collided. Protocols for scheduling the retransmission of collided messages are described below.

16.2

CAPACITY OF MULTIPLE ACCESS METHODS

It is interesting to compare FDMA, TDMA, and CDMA in terms of the information rate that each multiple access method achieves in an ideal AWGN channel of bandwidth W. Let us compare the capacity of K users, where each user has an average power $P_i = P$, for all $1 \leq i \leq K$. Recall that in an ideal band-limited AWGN channel of bandwidth W, the capacity of a single user is

$$C = W \log_2 \left(1 + \frac{P}{WN_0}\right) \tag{16.2–1}$$

where $\frac{1}{2}N_0$ is the power spectral density of the additive noise.

In FDMA, each user is allocated a bandwidth W/K. Hence, the capacity of each user is

$$C_K = \frac{W}{K} \log_2 \left[1 + \frac{P}{(W/K)N_0}\right] \tag{16.2–2}$$

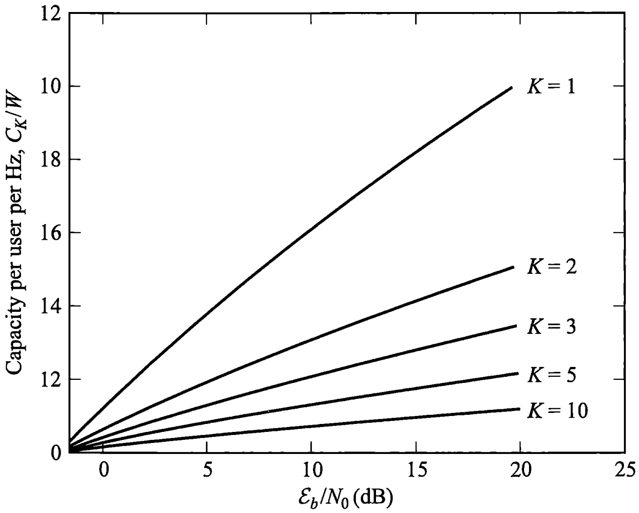


FIGURE 16.2–1
Normalized capacity as a function of $\mathcal{E}_b/N_0$ for FDMA.

and the total capacity for the K users is

$$KC_K = W \log_2 \left(1 + \frac{KP}{WN_0}\right) \tag{16.2–3}$$

Therefore, the total capacity is equivalent to that of a single user with average power $P_{\text{av}} = KP$.

It is interesting to note that for a fixed bandwidth W, the total capacity goes to infinity as the number of users increases linearly with K. On the other hand, as K increases, each user is allocated a smaller bandwidth (W/K) and, consequently, the capacity per user decreases. Figure 16.2–1 illustrates the capacity C_K per user normalized by the channel bandwidth W, as a function of $\mathcal{E}_b/N_0$, with K as a parameter. This expression is given as

$$\frac{C_K}{W} = \frac{1}{K} \log_2 \left[1 + K\frac{C_K}{W}\left(\frac{\mathcal{E}_b}{N_0}\right)\right] \tag{16.2–4}$$

A more compact form of Equation 16.2–4 is obtained by defining the normalized total capacity $C_n = KC_K/W$, which is the total bit rate for all K users per unit of bandwidth. Thus, Equation 16.2–4 may be expressed as

$$C_n = \log_2 \left(1 + C_n \frac{\mathcal{E}_b}{N_0}\right) \tag{16.2–5}$$

or, equivalently,

$$\frac{\mathcal{E}_b}{N_0} = \frac{2^{C_n} - 1}{C_n} \tag{16.2–6}$$

The graph of C_n versus $\mathcal{E}_b/N_0$ is shown in Figure 16.2–2. We observe that C_n increases as $\mathcal{E}_b/N_0$ increases above the minimum value of ln 2.

In a TDMA system, each user transmits for $1/K$ of the time through the channel of bandwidth W, with average power KP. Therefore, the capacity per user is

$$C_K = \left(\frac{1}{K}\right) W \log_2 \left(1 + \frac{KP}{WN_0}\right) \tag{16.2–7}$$

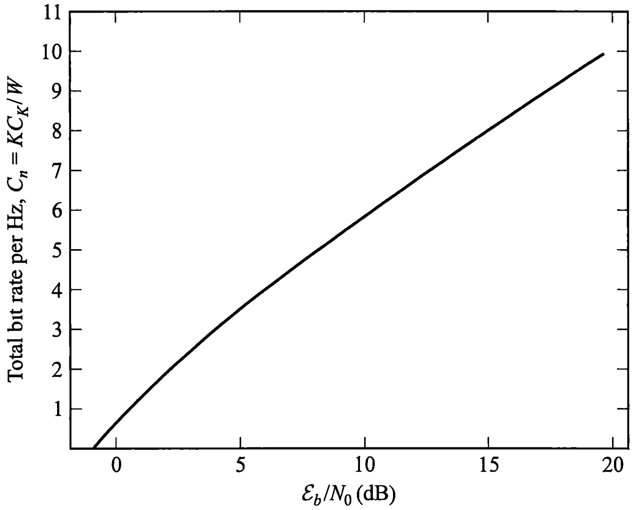


FIGURE 16.2–2
Total capacity per hertz as a function of $\mathcal{E}_b/N_0$ for FDMA.

which is identical to the capacity of an FDMA system. However, from a practical standpoint, we should emphasize that, in TDMA, it may not be possible for the transmitters to sustain a transmitter power of KP when K is very large. Hence, there is a practical limit beyond which the transmitter power cannot be increased as K is increased.

In a CDMA system, each user transmits a pseudorandom signal of a bandwidth W and average power P. The capacity of the system depends on the level of cooperation among the K users. At one extreme is noncooperative CDMA, in which the receiver for each user signal does not know the codes and spreading waveforms of the other users, or chooses to ignore them in the demodulation process. Hence, the other users' signals appear as interference at the receiver of each user. In this case, the multiuser receiver consists of a bank of K single-user matched filters. This is called *single-user detection*. If we assume that each user's pseudorandom signal waveform is Gaussian, then each user signal is corrupted by Gaussian interference of power $(K-1)P$ and additive Gaussian noise of power WN_0. Therefore, the capacity per user for single-user detection is

$$C_K = W\log_2\left[1 + \frac{P}{WN_0 + (K-1)P}\right] \tag{16.2–8}$$

or, equivalently,

$$\frac{C_K}{W} = \log_2\left[1 + \frac{C_K}{W}\frac{\mathcal{E}_b/N_0}{1 + (K-1)(C_K/W)(\mathcal{E}_b/N_0)}\right] \tag{16.2–9}$$

Figure 16.2–3 illustrates the graph of C_K/W versus $\mathcal{E}_b/N_0$, with K as a parameter.

For a large number of users, we may use the approximation $\ln(1+x) \le x$. Hence,

$$\frac{C_K}{W} \le \frac{C_K}{W}\frac{\mathcal{E}_b/N_0}{1 + K(C_K/W)(\mathcal{E}_b/N_0)}\log_2 e \tag{16.2–10}$$

or, equivalently, the normalized total capacity $C_n = KC_K/W$ is

$$\begin{aligned} C_n &\le \log_2 e - \frac{1}{\mathcal{E}_b/N_0} \\ &\le \frac{1}{\ln 2} - \frac{1}{\mathcal{E}_b/N_0} < \frac{1}{\ln 2} \end{aligned} \tag{16.2–11}$$

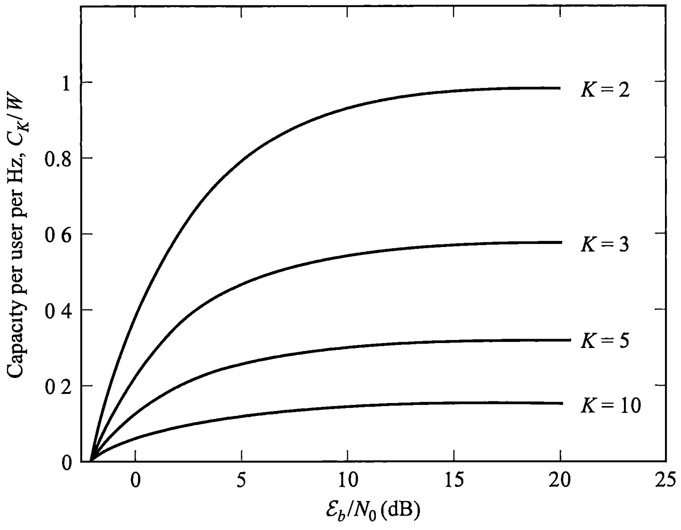


FIGURE 16.2–3
Normalized capacity as a function of $\mathcal{E}_b/N_0$ for noncooperative CDMA.

In this case, we observe that the total capacity does not increase with K as in TDMA and FDMA.

On the other hand, suppose that the K users cooperate by transmitting their coded signals synchronously in time, and the multiuser receiver jointly demodulates and decodes all the users' signals. This is called *multiuser detection and decoding*. Each user is assigned a rate $R_i, 1 \leq i \leq K$, and a code book containing a set of 2^{nR_i} codewords of power P. In each signal interval, each user selects an arbitrary codeword, say $\mathbf{X}_i$, from its own code book, and all users transmit their codewords simultaneously. Thus, the decoder at the receiver observes

$$Y = \sum_{i=1}^{K} X_i + Z \tag{16.2–12}$$

where Z is an additive noise vector. The optimum decoder looks for the K codewords, one from each code book, that have a vector sum closest to the received vector Y in Euclidean distance.

The achievable K-dimensional rate region for the K users in an AWGN channel, assuming equal power for each user, is given by the following equations:

$$R_i < W \log_2 \left(1 + \frac{P}{WN_0}\right), \qquad 1 \leq i \leq K \tag{16.2–13}$$

$$R_i + R_j < W \log_2 \left(1 + \frac{2P}{WN_0}\right), \qquad 1 \leq i, j \leq K \tag{16.2–14}$$

$$\vdots$$

$$R_{\text{SUM}}^{\text{MU}} = \sum_{i=1}^{K} R_i < W \log_2 \left(1 + \frac{KP}{WN_0}\right) \tag{16.2–15}$$

where $R_{\text{SUM}}^{\text{MU}}$ is the total (sum) rate achieved by the K users by employing multiuser detection. In the special case when all the rates are identical, the inequality 16.2–15 is dominant over the other $K-1$ inequalities. It follows that if the rates $\{R_i, 1 \le i \le K\}$ for the K cooperative synchronous users are selected to fall in the capacity region specified by the inequalities given above, then the probabilities of error for the K users tend to zero as the code block length n tends to infinity.

From the above discussion, we conclude that the sum of the rates of the K users $R_{\text{SUM}}^{\text{MU}}$ goes to infinity with K. Therefore, with coded synchronous transmission and joint detection and decoding, the capacity of CDMA has a form similar to that of FDMA and TDMA. Note that if all the rates in the CDMA system are selected to be identical to R, then Equation 16.2–15 reduces to

$$R < \frac{W}{K}\log_2\left(1+\frac{KP}{WN_0}\right) \tag{16.2–16}$$

which is the highest possible rate and is identical to the rate constraint in FDMA and TDMA. In this case, CDMA does not yield a higher rate than TDMA and FDMA. However, if the rates of the K users are selected to be unequal such that the inequalities 16.2–13 to 16.2–15 are satisfied, then it is possible to find the points in the achievable rate region such that the sum of the rates for the K users in CDMA exceeds the capacity of FDMA and TDMA.

EXAMPLE 16.2–1. Consider the case of two users in a CDMA system that employs coded signals as described above. The rates of the two users must satisfy the inequalities

$$R_1 < W\log_2\left(1+\frac{P}{WN_0}\right) \tag{16.2–17}$$

$$R_2 < W\log_2\left(1+\frac{P}{WN_0}\right) \tag{16.2–18}$$

$$R_1 + R_2 < W\log_2\left(1+\frac{2P}{WN_0}\right) \tag{16.2–19}$$

where P is the average transmitted power of each user and W is the signal bandwidth.

The capacity region for the two-user CDMA system with coded signal waveforms has the form illustrated in Figure 16.2–4, where

$$C_i = W\log_2\left(1+\frac{P_i}{WN_0}\right), \qquad i = 1, 2$$

are the capacities corresponding to the two users with $P_1 = P_2 = P$. We note that if user 1 is transmitting at capacity C_1, user 2 can transmit up to a maximum rate

$$\begin{aligned} R_{2m} &= W\log_2\left(1+\frac{2P}{WN_0}\right) - C_1 \\ &= W\log_2\left(1+\frac{P}{P+WN_0}\right) \end{aligned} \tag{16.2–20}$$

which is illustrated in Figure 16.2–4 as point A. This result has an interesting interpretation. We note that rate R_{2m} corresponds to the case in which the signal from user 1 is

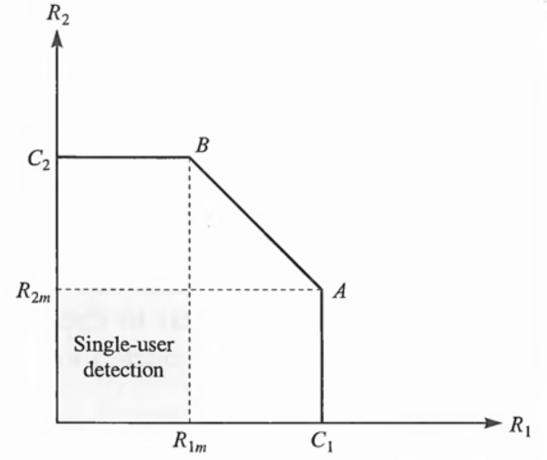


FIGURE 16.2–4
Capacity region of two-user CDMA multiple access Gaussian channel.

considered as an equivalent additive noise in the detection of the signal of user 2. On the other hand, user 1 can transmit at capacity C_1, since the receiver knows the transmitted signal from user 2 and, hence, it can eliminate its effect in detecting the signal of user 1.

Because of symmetry, a similar situation exists if user 2 is transmitting at capacity C_2. Then user 1 can transmit up to a maximum rate $R_{1m} = R_{2m}$, which is illustrated in Figure 16.2–4 as point B. In this case, we have a similar interpretation as above, with an interchange in the roles of user 1 and user 2.

The points A and B are connected by a straight line, which is defined by Equation 16.2–19. It is easily seen that this straight line is the boundary of the achievable rate region, since any point on the line corresponds to the maximum rate $W \log_2 (1 + 2P/WN_0)$, which can be obtained by simply time sharing the channel between the two users.

In the next section, we consider the problem of signal detection for a multiuser CDMA system and assess the performance and the computational complexity of several receiver structures.

16.3
MULTIUSER DETECTION IN CDMA SYSTEMS

As we have observed, TDMA and FDMA are multiple access methods in which the channel is partitioned into independent, single-user subchannels, i.e., non-overlapping time slots or frequency bands, respectively. In CDMA, each user is assigned a distinct signature sequence (or waveform), which the user employs to modulate and spread the information-bearing signal. The signature sequences also allow the receiver to demodulate the message transmitted by multiple users of the channel, who transmit simultaneously and, generally, asynchronously.

In this section, we treat the demodulation and detection of multiuser uncoded CDMA signals. We shall see that the optimum maximum-likelihood detector has a computational complexity that grows exponentially with the number of users. Such a high complexity serves as a motivation to devise suboptimum detectors having lower computational complexities. Finally, we consider the performance characteristics of the various detectors.

16.3–1 CDMA Signal and Channel Models

Let us consider a CDMA channel that is shared by K simultaneous users. Each user is assigned a signature waveform $g_k(t)$ of duration T, where T is the symbol interval. A signature waveform may be expressed as

$$g_k(t) = \sum_{n=0}^{L-1} a_k(n) p(t - nT_c), \qquad 0 \le t \le T \tag{16.3–1}$$

where $\{a_k(n), 0 \le n \le L - 1\}$ is a pseudonoise (PN) code sequence consisting of L chips that take values $\{\pm 1\}$, $p(t)$ is a pulse of duration T_c, and T_c is the chip interval. Thus, we have L chips per symbol and $T = LT_c$. Without loss of generality, we assume that all K signature waveforms have unit energy, i.e.,

$$\int_0^T g_k^2(t)\, dt = 1 \tag{16.3–2}$$

The cross correlations between pairs of signature waveforms play an important role in the metrics for the signal detector and on its performance. We define the following cross correlations, where $0 \le \tau \le T$ and $i < j$,

$$\rho_{ij}(\tau) = \int_\tau^T g_i(t) g_j(t - \tau)\, dt \tag{16.3–3}$$

$$\rho_{ji}(\tau) = \int_0^\tau g_i(t) g_j(t + T + \tau)\, dt \tag{16.3–4}$$

The cross correlations in Equations 16.3–3 and 16.3–4 apply to asynchronous transmissions among the K users. For synchronous transmission, we need only $\rho_{ij}(0)$.

For simplicity, we assume that binary antipodal signals are used to transmit the information from each user. Hence, let the information sequence of the kth user be denoted by $\{b_k(m)\}$, where the value of each information bit may be ± 1. It is convenient to consider the transmission of a block of bits of some arbitrary length, say N. Then, the data block from the kth user is

$$\boldsymbol{b}_k = [b_k(1) \cdots b_k(N)]^t \tag{16.3–5}$$

and the corresponding equivalent lowpass, transmitted waveform may be expressed as

$$s_k(t) \doteq \sqrt{\mathcal{E}_k} \sum_{i=1}^{N} b_k(i) g_k(t - iT) \tag{16.3–6}$$

where $\mathcal{E}_k$ is the signal energy per bit. The composite transmitted signal for the K users may be expressed as

$$\begin{aligned} s(t) &= \sum_{k=1}^{K} s_k(t - \tau_k) \\ &= \sum_{k=1}^{K} \sqrt{\mathcal{E}_k} \sum_{i=1}^{N} b_k(i) g_k(t - iT - \tau_k) \end{aligned} \tag{16.3–7}$$

where $\{\tau_k\}$ are the transmission delays, which satisfy the condition $0 \le \tau_k < T$ for $1 \le k \le K$. Without loss of generality, we assume that $0 \le \tau_1 \le \tau_2 \le \cdots \le \tau_K < T$. This is the model for the multiuser transmitted signal in an asynchronous mode. In the special case of synchronous transmission, $\tau_k = 0$ for $1 \le k \le K$.

The transmitted signal is assumed to be corrupted by AWGN. Hence, the received signal may be expressed as

$$r(t) = s(t) + n(t) \tag{16.3–8}$$

where $s(t)$ is given by Equation 16.3–7 and $n(t)$ is the noise, with power spectral density $\frac{1}{2}N_0$.

16.3–2 The Optimum Multiuser Receiver

The optimum receiver is defined as the receiver that selects the most probable sequence of bits $\{b_k(n), 1 \le n \le N, 1 \le k \le K\}$ given the received signal $r(t)$ observed over the time interval $0 \le t \le NT + 2T$. First, let us consider the case of synchronous transmission; later, we shall consider asynchronous transmission.

Synchronous transmission In synchronous transmission, each (user) interferer produces exactly one symbol which interferes with the desired symbol. In additive white Gaussian noise, it is sufficient to consider the signal received in one signal interval, say $0 \le t \le T$, and determine the optimum receiver. Hence, $r(t)$ may be expressed as

$$r(t) = \sum_{k=1}^{K} \sqrt{\mathcal{E}_k} b_k(1) g_k(t) + n(t), \qquad 0 \le t \le T \tag{16.3–9}$$

The optimum maximum-likelihood receiver computes the log-likelihood function

$$\Lambda(\boldsymbol{b}) = \int_0^T \left[r(t) - \sum_{k=1}^{K} \sqrt{\mathcal{E}_k} b_k(1) g_k(t) \right]^2 dt \tag{16.3–10}$$

and selects the information sequence $\{b_k(1), 1 \le k \le K\}$ that minimizes $\Lambda(\boldsymbol{b})$. If we expand the integral in Equation 16.3–10, we obtain

$$\begin{aligned} \Lambda(\boldsymbol{b}) = & \int_0^T r^2(t)\,dt - 2\sum_{k=1}^{K} \sqrt{\mathcal{E}_k} b_k(1) \int_0^T r(t) g_k(t)\,dt \\ & + \sum_{j=1}^{K}\sum_{k=1}^{K} \sqrt{\mathcal{E}_j \mathcal{E}_k} b_k(1) b_j(1) \int_0^T g_k(t) g_j(t)\,dt \end{aligned} \tag{16.3–11}$$

We observe that the integral involving $r^2(t)$ is common to all possible sequences $\{b_k(1)\}$ and is of no relevance in determining which sequence was transmitted. Hence, it may be neglected. The term

$$r_k = \int_0^T r(t) g_k(t)\, dt, \qquad 1 \le k \le K \tag{16.3–12}$$

represents the cross correlation of the received signal with each of the K signature sequences. Instead of cross correlators, we may employ matched filters. Finally, the integral involving $g_k(t)$ and $g_j(t)$ is simply

$$\rho_{jk}(0) = \int_0^T g_j(t) g_k(t)\, dt \tag{16.3–13}$$

Therefore, Equation 16.3–11 may be expressed in the form of correlation metrics

$$C(\boldsymbol{r}_K, \boldsymbol{b}_K) = 2\sum_{k=1}^{K} \sqrt{\mathcal{E}_k} b_k(1) r_k - \sum_{j=1}^{K}\sum_{k=1}^{K} \sqrt{\mathcal{E}_j \mathcal{E}_k} b_k(1) b_j(1) \rho_{jk}(0) \tag{16.3–14}$$

These correlation metrics may also be expressed in vector inner product form as

$$C(\boldsymbol{r}_K, \boldsymbol{b}_K) = 2\boldsymbol{b}_K^t \boldsymbol{r}_K - \boldsymbol{b}_K^t \boldsymbol{R}_s \boldsymbol{b}_K \tag{16.3–15}$$

where

$$\boldsymbol{r}_K = [r_1 \quad r_2 \quad \cdots \quad r_K]^t, \qquad \boldsymbol{b}_K = [\sqrt{\mathcal{E}_1} b_1(1) \ldots \sqrt{\mathcal{E}_K} b_K(1)]^t$$

and $\boldsymbol{R}_s$ is the correlation matrix, with elements $\rho_{jk}(0)$. It is observed that the optimum detector must have knowledge of the received signal energies in order to compute the correlation metrics. Figure 16.3–1 depicts the optimum multiuser receiver.

There are 2^K possible choices of the bits in the information sequence of the K users. The optimum detector computes the correlation metrics for each sequence and selects the sequence that yields the largest correlation metric. We observe that the optimum detector has a complexity that grows exponentially with the number of users, K.

In summary, the optimum receiver for symbol-synchronous transmission consists of a bank of K correlators or matched filters followed by a detector that computes the 2^K correlation metrics given by Equation 16.3–15 corresponding to the 2^K possible transmitted information sequences. Then, the detector selects the sequence corresponding to the largest correlation metric.

Asynchronous transmission In this case, there are exactly two consecutive symbols from each interferer that overlap a desired symbol. We assume that the receiver knows the received signal energies $\{\mathcal{E}_k\}$ for the K users and the transmission delays $\{\tau_k\}$. Clearly, these parameters must be measured at the receiver or provided to the receiver as side information by the users via some control channel.

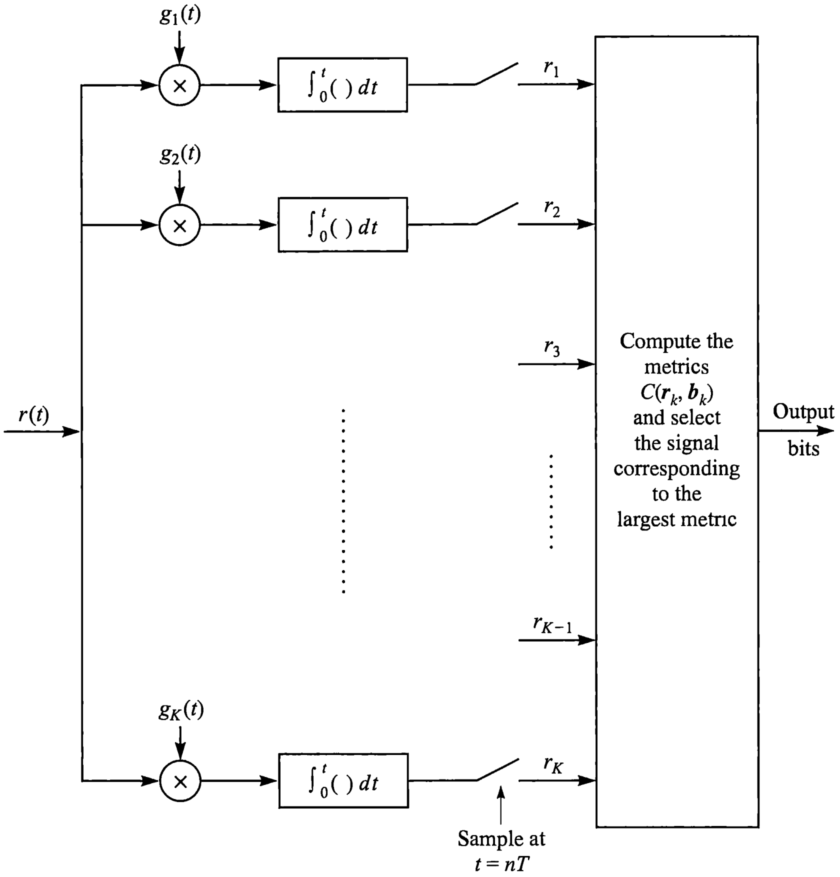

$g_1(t)$
$\int_0^t (\,)\,dt$
r_1
$g_2(t)$
$\int_0^t (\,)\,dt$
r_2
r_3
$r(t)$
Compute the metrics $C(r_k, b_k)$ and select the signal corresponding to the largest metric
Output bits
r_{K-1}
$g_K(t)$
$\int_0^t (\,)\,dt$
r_K
Sample at $t = nT$

FIGURE 16.3–1
Optimum multiuser receiver for synchronous transmission.

The optimum maximum-likelihood receiver computes the log-likelihood function

$$\begin{aligned}\Lambda(\boldsymbol{b}) &= \int_0^{NT+2T}\left[r(t) - \sum_{k=1}^{K}\sqrt{\mathcal{E}_k}\sum_{i=1}^{N} b_k(i) g_k(t - iT - \tau_k)\right]^2 dt \\ &= \int_0^{NT+2T} r^2(t)\,dt - 2\sum_{k=1}^{K}\sqrt{\mathcal{E}_k}\sum_{i=1}^{N} b_k(i)\int_0^{NT+2T} r(t) g_k(t - iT - \tau_k)\,dt \\ &\quad + \sum_{k=1}^{K}\sum_{l=1}^{K}\sqrt{\mathcal{E}_k\mathcal{E}_l}\sum_{i=1}^{N}\sum_{j=1}^{N} b_k(i) b_l(j)\int_0^{NT+2T} g_k(t - iT - \tau_k) g_l(t - jT - \tau_l)\,dt\end{aligned} \tag{16.3–16}$$

where $\boldsymbol{b}$ represents the data sequences from the K users. The integral involving $r^2(t)$ may be ignored, since it is common to all possible information sequences. The integral

$$r_k(i) \equiv \int_{iT+\tau_k}^{(i+1)T+\tau_k} r(t) g_k(t - iT - \tau_k)\,dt, \qquad 1 \le i \le N \tag{16.3–17}$$

represents the outputs of the correlator or matched filter for the kth user in each of the signal intervals. Finally, the integral

$$\int_0^{NT+2T} g_k(t - iT - \tau_k) g_l(t - jT - \tau_l)\, dt$$
$$= \int_{-iT-\tau_k}^{NT+2T-iT-\tau_k} g_k(t) g_l(t + iT - jT + \tau_k - \tau_1)\, dt \tag{16.3–18}$$

may be easily decomposed into terms involving the cross correlation $\rho_{kl}(\tau) = \rho_{kl}(\tau_l - \tau_k)$ for $k \le 1$ and $\rho_{ik}(\tau)$ for $k > 1$. Therefore, we observe that the log-likelihood function may be expressed in terms of a correlation metric that involves the outputs $\{r_k(i), 1 \le k \le K, \le i \le N\}$ of K correlators or matched filters—one for each of the K signature sequences. Using vector notation, it can be shown that the NK correlator or matched filter outputs $\{r_k(i)\}$ can be expressed in the form

$$\boldsymbol{r} = \boldsymbol{R}_N \boldsymbol{b} + \boldsymbol{n} \tag{16.3–19}$$

where, by definition

$$\begin{aligned} \boldsymbol{r} &= [\boldsymbol{r}^t(1) \quad \boldsymbol{r}^t(2) \quad \cdots \quad \boldsymbol{r}^t(N)]^t \\ \boldsymbol{r}(i) &= [r_1(i) \quad r_2(i) \quad \cdots \quad r_K(i)]^t \end{aligned} \tag{16.3–20}$$

$$\begin{aligned} \boldsymbol{b} &= [\boldsymbol{b}^t(1) \quad \boldsymbol{b}^t(2) \quad \cdots \quad \boldsymbol{b}^t(N)]^t \\ \boldsymbol{b}(i) &= [\sqrt{\mathcal{E}_1} b_1(i) \quad \sqrt{\mathcal{E}_2} b_2(i) \quad \cdots \quad \sqrt{\mathcal{E}_K} b_K(i)]^t \end{aligned} \tag{16.3–21}$$

$$\begin{aligned} \boldsymbol{n} &= [\boldsymbol{n}^t(1) \quad \boldsymbol{n}^t(2) \quad \cdots \quad \boldsymbol{n}^t(N)]^t \\ \boldsymbol{n}(i) &= [n_1(i) \quad n_2(i) \quad \cdots \quad n_K(i)]^t \end{aligned} \tag{16.3–22}$$

$$\boldsymbol{R}_N = \begin{bmatrix} \boldsymbol{R}_a(0) & \boldsymbol{R}_a^t(1) & \boldsymbol{0} & \cdots & \cdots & \boldsymbol{0} \\ \boldsymbol{R}_a(1) & \boldsymbol{R}_a(0) & \boldsymbol{R}_a^t(1) & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{R}_a(1) & \boldsymbol{R}_a(0) & \boldsymbol{R}_a^t(1) \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{R}_a(1) & \boldsymbol{R}_a(0) \end{bmatrix} \tag{16.3–23}$$

and $\boldsymbol{R}_a(m)$ is a $K \times K$ matrix with elements

$$R_{kl}(m) = \int_{-\infty}^{\infty} g_k(t - \tau_k) g_l(t + mT - \tau_l)\, dt \tag{16.3–24}$$

The Gaussian noise vectors $\boldsymbol{n}(i)$ have zero-mean and autocorrelation matrix

$$E[\boldsymbol{n}(k)\boldsymbol{n}^t(j)] = \tfrac{1}{2} N_0 \boldsymbol{R}_a(k - j) \tag{16.3–25}$$

Note that the vector $\boldsymbol{r}$ given by Equation 16.3–19 constitutes a set of sufficient statistics for estimating the transmitted bits $b_k(i)$.

If we adopt a block processing approach, the optimum ML detector must compute 2^{NK} correlation metrics and select the K sequences of length N that correspond to the largest correlation metric. Clearly, such an approach is much too complex computationally to be implemented in practice, especially when K and N are large. An alternative approach is ML sequence estimation employing the Viterbi algorithm. In order to construct a sequential-type detector, we make use of the fact that each transmitted symbol overlaps at most with $2K-2$ symbols. Thus, a significant reduction in computational complexity is obtained with respect to the block size parameter N, but the exponential dependence on K cannot be reduced.

It is apparent that the optimum ML receiver employing the Viterbi algorithm involves such a high computational complexity that its use in practice is limited to communication systems where the number of users is extremely small, e.g., $K<10$. For larger values of K, one may consider a sequential-type detector that is akin to either the sequential decoding or the stack algorithms described in Chapter 8. Below, we consider a number of suboptimum detectors whose complexity grows linearly with K.

16.3–3 Suboptimum Detectors

In the above discussion, we observed that the optimum detector for the K CDMA users has a computational complexity, measured in the number of arithmetic operations (additions and multiplications/divisions) per modulated symbol, that grows exponentially with K. In this subsection we describe suboptimum detectors with computational complexities that grow linearly with the number of users, K. We begin with the simplest suboptimum detector, which we call the conventional (single-user) detector.

Conventional single-user detector In conventional single-user detection, the receiver for each user consists of a demodulator that correlates (or match-filters) the received signal with the signature sequence of the user and passes the correlator output to the detector, which makes a decision based on the single correlator output. Thus, the conventional detector neglects the presence of the other users of the channel or, equivalently, assumes that the aggregate noise plus interference is white and Gaussian.

Let us consider synchronous transmission. Then, the output of the correlator for the kth user for the signal in the interval $0 \le t \le T$ is

$$r_k = \int_0^T r(t) g_k(t)\, dt \tag{16.3–26}$$

$$= \sqrt{\mathcal{E}_k} b_k(1) + \sum_{\substack{j=1 \\ j\neq k}}^{K} \sqrt{\mathcal{E}_j} b_j(1) \rho_{jk}(0) + n_k(1) \tag{16.3–27}$$

where the noise component $n_k(1)$ is given as

$$n_k(1) = \int_0^T n(t) g_k(t)\, dt \tag{16.3–28}$$

Since $n(t)$ is white Gaussian noise with power spectral density $\frac{1}{2}N_0$, the variance of $n_k(1)$ is

$$E\left[n_k^2(1)\right] = \tfrac{1}{2}N_0 \int_0^T g_k^2(t)\,dt = \tfrac{1}{2}N_0 \tag{16.3–29}$$

Clearly, if the signature sequences are orthogonal, the interference from the other users given by the middle term in Equation 16.3–27 vanishes and the conventional single-user detector is optimum. On the other hand, if one or more of the other signature sequences are not orthogonal to the user signature sequence, the interference from the other users can become excessive if the power levels of the signals (or the received signal energies) of one or more of the other users is sufficiently larger than the power level of the kth user. This situation is generally called the *near–far problem* in multiuser communications, and necessitates some type of power control for conventional detection.

In asynchronous transmission, the conventional detector is more vulnerable to interference from other users. This is because it is not possible to design signature sequences for any pair of users that are orthogonal for all time offsets. Consequently, interference from other users is unavoidable in asynchronous transmission with the conventional single-user detection. In such a case, the near–far problem resulting from unequal power in the signals transmitted by the various users is particularly serious. The practical solution generally requires a power adjustment method that is controlled by the receiver via a separate communication channel that all users are continuously monitoring. Another option is to employ one of the multiuser detectors described below.

Decorrelating detector We observe that the conventional detector has a complexity that grows linearly with the number of users, but its vulnerability to the near–far problem requires some type of power control. We shall now devise another type of detector that also has a linear computational complexity but does not exhibit the vulnerability to other-user interference.

Let us first consider the case of symbol-synchronous transmission. In this case, the received signal vector $\boldsymbol{r}_K$ that represents the output of the K matched filters is

$$\boldsymbol{r}_K = \boldsymbol{R}_s \boldsymbol{b}_K + \boldsymbol{n}_K \tag{16.3–30}$$

where $\boldsymbol{b}_K = [\sqrt{\mathcal{E}_1}b_1(1)\quad \sqrt{\mathcal{E}_2}b_2(1)\quad \cdots \quad \sqrt{\mathcal{E}_K}b_K(1)]^t$ and the noise vector with elements $\boldsymbol{n}_K = [n_1(1)\ \ n_2(1)\cdots n_K(1)]^t$ has a covariance

$$E\left(\boldsymbol{n}_K \boldsymbol{n}^t{}_K\right) = \frac{N_0}{2}\boldsymbol{R}_s \tag{16.3–31}$$

Since the noise is Gaussian, $\boldsymbol{r}_K$ is described by a K-dimensional Gaussian PDF with mean $\boldsymbol{R}_s\boldsymbol{b}_K$ and covariance $\boldsymbol{R}_s$. That is,

$$p(\boldsymbol{r}_K|\boldsymbol{b}_K) = \frac{1}{\sqrt{(N_0\pi)^K \det \boldsymbol{R}_s}} \exp\left[-\frac{1}{N_0}(\boldsymbol{r}_K - \boldsymbol{R}_s\boldsymbol{b}_K)^t \boldsymbol{R}_s^{-1}(\boldsymbol{r}_K - \boldsymbol{R}_s\boldsymbol{b}_K)\right] \tag{16.3–32}$$

The best linear estimate of $\boldsymbol{b}_K^0$ is the value of $\boldsymbol{b}_K$ that minimizes the likelihood function

$$\Lambda(\boldsymbol{b}_K) = (\boldsymbol{r}_K - \boldsymbol{R}_s\boldsymbol{b}_K)^t \boldsymbol{R}_s^{-1}(\boldsymbol{r}_K - \boldsymbol{R}_s\boldsymbol{b}_K) \tag{16.3–33}$$

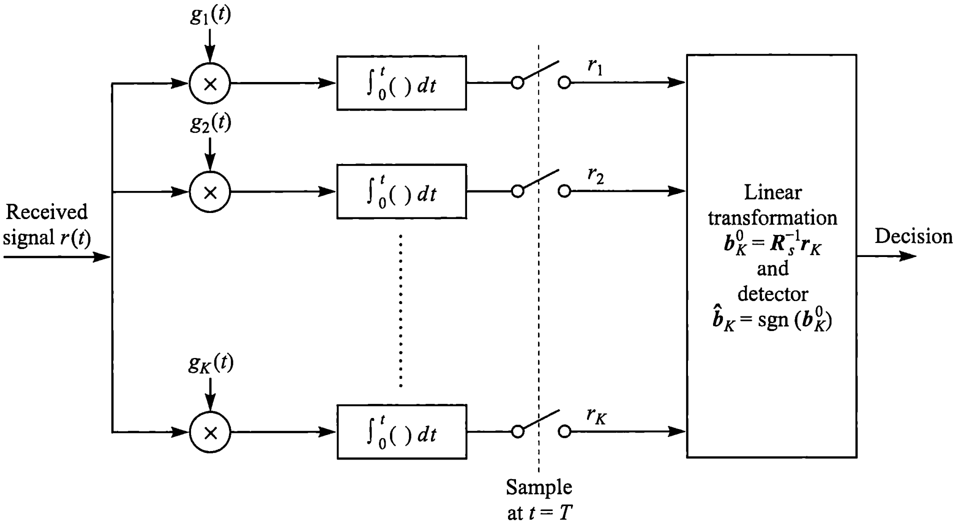

$g_1(t)$
$g_2(t)$
$g_K(t)$
Received signal $r(t)$
$\int_0^t (\,)\,dt$
r_1
r_2
r_K
Sample at $t = T$
Linear transformation
$b_K^0 = R_s^{-1} r_K$
and
detector
$\hat{b}_K = \mathrm{sgn}(b_K^0)$
Decision

FIGURE 16.3–2
Receiver structure for decorrelation receiver.

The result of this minimization yields

$$b_k^0 = R_s^{-1} r_K \tag{16.3–34}$$

Then, the detected symbols are obtained by taking the sign of each element of b_K^0, i.e.,

$$\hat{b}_K = \mathrm{sgn}(b_K^0) \tag{16.3–35}$$

Figure 16.3–2 illustrates the receiver structure. Since the estimate b_K^0 is obtained by performing a linear transformation on the vector of correlator outputs, the computational complexity is linear in K.

The reader should observe that the best (maximum-likelihood) linear estimate of b_K given by Equation 16.3–34 is different from the optimum non-linear ML sequence detector that finds the best discrete-valued $\{\pm 1\}$ sequence that maximizes the likelihood function. It is also interesting to note that the estimate b_K^0 is the best linear estimate that maximizes the correlation metric given by Equation 16.3–15.

An interesting interpretation of the detector that computes b_K^0 as in Equation 16.3–34 and makes decisions according to Equation 16.3–35 is obtained by considering the case of $K = 2$ users. In this case,

$$R_s = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \tag{16.3–36}$$

$$R_s^{-1} = \frac{1}{1-\rho^2} \begin{bmatrix} 1 & -\rho \\ -\rho & 1 \end{bmatrix} \tag{16.3–37}$$

where

$$\rho = \int_0^T g_1(t) g_2(t)\, dt \tag{16.3–38}$$

Then, if we correlate the received signal

$$r(t) = \sqrt{\mathcal{E}_1} b_1 g_1(t) + \sqrt{\mathcal{E}_2} b_2 g_2(t) + n(t) \tag{16.3–39}$$

with $g_1(t)$ and $g_2(t)$, we obtain

$$\boldsymbol{r}_2 = \begin{bmatrix} \sqrt{\mathcal{E}_1} b_1 + \rho\sqrt{\mathcal{E}_2} b_2 + n_1 \\ \rho\sqrt{\mathcal{E}_1} b_1 + \sqrt{\mathcal{E}_2} b_2 + n_2 \end{bmatrix} \tag{16.3–40}$$

where n_1 and n_2 are the noise components at the output of the correlators. Therefore,

$$\begin{aligned} \boldsymbol{b}_2^0 &= \boldsymbol{R}_s^{-1}\boldsymbol{r}_2 \\ &= \begin{bmatrix} \sqrt{\mathcal{E}_1} b_1 + (n_1 - \rho n_2)/(1-\rho^2) \\ \sqrt{\mathcal{E}_2} b_2 + (n_2 - \rho n_1)/(1-\rho^2) \end{bmatrix} \end{aligned} \tag{16.3–41}$$

This is a very interesting result, because the transformation $\boldsymbol{R}_s^{-1}$ has eliminated the interference components between the two users. Consequently, the near–far problem is eliminated and there is no need for power control.

It is interesting to note that a result similar to Equation 16.3–41 is obtained if we correlate $r(t)$ given by Equation 16.3–39 with two modified signature waveforms

$$g_1'(t) = g_1(t) - \rho g_2(t) \tag{16.3–42}$$

$$g_2'(t) = g_2(t) - \rho g_1(t) \tag{16.3–43}$$

This means that, by correlating the received signal with the modified signature waveforms, we have tuned out or *decorrelated* the multiuser interference. Hence, the detector based on Equation 16.3–34 is called a *decorrelating detector.*

In asynchronous transmission, the received signal at the output of the correlators is given by Equation 16.3–19. Hence, the log-likelihood function is given as

$$\Lambda(\boldsymbol{b}) = (\boldsymbol{r} - \boldsymbol{R}_N\boldsymbol{b})^t \boldsymbol{R}_N^{-1}(\boldsymbol{r} - \boldsymbol{R}_N\boldsymbol{b}) \tag{16.3–44}$$

where $\boldsymbol{R}_N$ is defined by Equation 16.3–23 and $\boldsymbol{b}$ is given by Equation 16.3–21. It is relatively easy to show that the vector $\boldsymbol{b}$ that minimizes $\Lambda(\boldsymbol{b})$ is

$$\boldsymbol{b}^0 = \boldsymbol{R}_N^{-1}\boldsymbol{r} \tag{16.3–45}$$

This is the ML estimate of $\boldsymbol{b}$ and it is again obtained by performing a linear transformation of the outputs from the bank of correlators of matched filters.

Since $\boldsymbol{r} = \boldsymbol{R}_N\boldsymbol{b} + \boldsymbol{n}$, it follows from Equation 16.3–45 that

$$\boldsymbol{b}^0 = \boldsymbol{b} + \boldsymbol{R}_N^{-1}\boldsymbol{n} \tag{16.3–46}$$

Therefore, $\boldsymbol{b}^0$ is an unbiased estimate of $\boldsymbol{b}$. This means that the multiuser interference has been eliminated, as in the case of symbol-synchronous transmission. Hence, this detector for asynchronous transmission is also called a *decorrelating detector.*

A computationally efficient method for obtaining the solution given by Equation 16.3–45 is the square-root factorization method described in Appendix D. Of course, there are many other methods that may be used to invert the matrix $\boldsymbol{R}_N$. Iterative methods to decorrelate the signals have also been explored.

Minimum mean-square-error detector In the above discussion, we showed that the linear ML estimate of $\boldsymbol{b}$ is obtained by minimizing the quadratic log-likelihood function in Equation 16.3–44. Thus, we obtained the result given by Equation 16.3–45, which is an estimate derived by performing a linear transformation on the outputs of the bank of correlators or matched filters.

Another, somewhat different, solution is obtained if we seek the linear transformation $\boldsymbol{b}^0 = \boldsymbol{A}\boldsymbol{r}$, where the matrix $\boldsymbol{A}$ is to be determined so as to minimize the mean square error (MSE)

$$\begin{aligned} J(\boldsymbol{b}) &= E[(\boldsymbol{b} - \boldsymbol{b}^0)^t(\boldsymbol{b} - \boldsymbol{b}^0)] \\ &= E[(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{r})^t(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{r})] \end{aligned} \tag{16.3–47}$$

where the expectation is with respect to the data vector $\boldsymbol{b}$ and the additive noise $\boldsymbol{n}$. The optimum matrix $\boldsymbol{A}$ may be found by forcing the error $(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{r})$ to be orthogonal to the data vector $\boldsymbol{r}$. Thus,

$$\begin{aligned} &E[(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{r})\boldsymbol{r}^t] = \boldsymbol{0} \\ &E(\boldsymbol{b}\boldsymbol{r}') - \boldsymbol{A}E(\boldsymbol{r}\boldsymbol{r}') = \boldsymbol{0} \end{aligned} \tag{16.3–48}$$

Let us consider the case of synchronous transmission. We have

$$E\left(\boldsymbol{b}_K\boldsymbol{r}_K^t\right) = E\left(\boldsymbol{b}_K\boldsymbol{b}_K^t\right)\boldsymbol{R}_s^t = \boldsymbol{D}\boldsymbol{R}_s^t \tag{16.3–49}$$

and

$$\begin{aligned} E\left(\boldsymbol{r}_K\boldsymbol{r}_K^t\right) &= E[(\boldsymbol{R}_s\boldsymbol{b}_K + \boldsymbol{n}_K)(\boldsymbol{R}_s\boldsymbol{b}_K + \boldsymbol{n}_K)^t] \\ &= \boldsymbol{R}_s\boldsymbol{D}\boldsymbol{R}_s^t + \frac{N_0}{2}\boldsymbol{R}_s^t \end{aligned} \tag{16.3–50}$$

where $\boldsymbol{D}$ is a diagonal matrix with diagonal elements $\{\mathcal{E}_k, 1 \leq k \leq K\}$. By substituting Equation 16.3–49 and 16.3–50 into Equation 16.3–48 and solving for $\boldsymbol{A}$, we obtain

$$\boldsymbol{A}^0 = \left(\boldsymbol{R}_s + \frac{N_0}{2}\boldsymbol{D}^{-1}\right)^{-1} \tag{16.3–51}$$

Then,

$$\boldsymbol{b}_K^0 = \boldsymbol{A}^0\boldsymbol{r}_K \tag{16.3–52}$$

and

$$\hat{\boldsymbol{b}}_K = \operatorname{sgn}\left(\boldsymbol{b}_K^0\right) \tag{16.3–53}$$

Similarly, for asynchronous transmission, it can be shown that the optimum choice of $\boldsymbol{A}$ that minimizes $J(\boldsymbol{b})$ is

$$\boldsymbol{A}^0 = \left(\boldsymbol{R}_N + \tfrac{1}{2}N_0\boldsymbol{I}\right)^{-1} \tag{16.3–54}$$

and, hence,

$$\boldsymbol{b}^0 = \left(\boldsymbol{R}_N + \tfrac{1}{2}N_0\boldsymbol{I}\right)^{-1}\boldsymbol{r} \tag{16.3–55}$$

The output of the detector is then $\hat{\boldsymbol{b}} = \operatorname{sgn}(\boldsymbol{b}^0)$.

The estimate given by Equation 16.3–52 or 16.3–55 is called the *minimum MSE* (MMSE) estimate of $\boldsymbol{b}$. Note that when $\frac{1}{2}N_0$ is small compared with the diagonal elements of $\boldsymbol{R}_N$, the MMSE solution approaches the ML solution given by Equation 16.3–45. On the other hand, when the noise level is large compared with the signal level in the diagonal elements of $\boldsymbol{R}_N$, $\boldsymbol{A}^0$ approaches the identity matrix (scaled by $\frac{1}{2}N_0$). In this low-SNR case, the detector basically ignores the interference from other users, because the additive noise is the dominant term. It should also be noted that the MMSE criterion produces a biased estimate of $\boldsymbol{b}$. Hence, there is some residual multiuser interference.

To perform the computations that lead to the values of $\boldsymbol{b}$, we solve the set of linear equations

$$\left(\boldsymbol{R}_N + \tfrac{1}{2}N_0\boldsymbol{I}\right)\boldsymbol{b} = \boldsymbol{r} \tag{16.3–56}$$

This solution may be computed efficiently using a square-root factorization of the matrix $\boldsymbol{R}_N + \frac{1}{2}N_0\boldsymbol{I}$ as indicated above. Thus, to detect NK bits requires $3NK^2$ multiplications. Therefore, the computational complexity is $3K$ multiplications per bit, which is independent of the block length N and is linear in K.

We observe that both the decorrelating detector and the MMSE detector exhibit the desirable property of being near-far resistant. In fact, in the case of the decorrelating detector, the interference from other users is completely eliminated.

We also observe that both the decorrelating detector and the MMSE detector described above involve performing linear transformations on a block of data obtained from K correlators or matched filters. The linear transformations are akin to the linear equalization of intersymbol interference treated in Chapter 9. In fact, the decorrelating detector is akin to the zero-forcing linear equalizer, and the MMSE detector is akin to the linear MMSE equalizer. Consequently, these multiuser detectors for asynchronous transmission can be implemented by employing a tapped-delay-line filter with adjustable coefficients for each user and selecting the filter coefficients to either eliminate the interuser interference or to minimize the MSE for each user signal. Thus, the received information bits are estimated sequentially with finite delay, instead of as a block.

A decision-feedback-type filter can be used instead of a linear filter to implement the multiuser detector that processes the data sequentially. In particular, Xie et al. (1990b) demonstrated that the transmitted bits may be recovered sequentially from the received signal by employing a form of a decision-feedback equalizer with finite delay. Hence, there is a similarity between the detection of signals corrupted by ISI in a single-user communication system and the detection of signals in a multiuser system with asynchronous transmission.

16.3–4 Successive Interference Cancellation

Another multiuser detection technique is called successive interference cancellation (SIC). This technique is based on removing the interfering signal waveforms from the received signal, one at a time as they are detected. One approach is to demodulate the users in the order of decreasing received powers. Thus, the user having the strongest received signal is demodulated first. After a signal has been demodulated and detected, the detected information is used to subtract the signal of the particular user from the received signal.

When making a decision about the transmitted information of the kth user, we assume that the decisions of users $k+1, \ldots, K$ are correct and neglect the presence of users $1, \ldots, k-1$. Therefore, the decision for the information bit of the kth user, for synchronous transmission, is

$$\hat{b}_k = \operatorname{sgn}\left[r_k - \sum_{j=k+1}^{K} \sqrt{\mathcal{E}_j}\,\rho_{jk}(0)\hat{b}_j\right] \tag{16.3–57}$$

where r_k is the output of the correlator or matched filter corresponding to the kth user's signature sequence.

The approach based on demodulating the user signals in the order of decreasing received powers does not take into account the cross correlations among users. An alternative approach is to demodulate the user signals according to the powers at the outputs of the cross correlators or matched filters, i.e., according to the correlation metrics

$$E\left\{\left[\int_0^T g_k(t)r(t)\,dt\right]^2\right\} = \mathcal{E}_k + \sum_{j\neq k} \mathcal{E}_j \rho_{jk}^2(0) + \frac{N_0}{2} \tag{16.3–58}$$

which applies to the case of synchronous transmission.

We make the following observations regarding the SIC of multiuser interference. First of all, SIC requires that we estimate the received signal powers of the users in order to cancel the interference. Estimation errors result in residual multiuser interference, which causes a degradation in performance. Secondly, the interference from users whose signals are weaker than the user signal being detected is treated as additive interference. Thirdly, the computational complexity in the demodulation of a user information bit is linear in the number of users. Finally, the delay in demodulating the weakest user increases linearly with the number of users.

SIC is easily generalized to asynchronous signal transmission. In this case, both the user signal strengths and the time delays must be estimated.

Finally, we note that the SIC multiuser detector given in Equation 16.3–57 is also a suboptimum detector, since the signals of weaker users are treated as additive interference. The jointly optimum interference canceller for synchronous transmission may be defined as the detector which computes the decisions $\hat{b}_k$ as

$$\hat{b}_k = \operatorname{sgn}\left[r_k - \sum_{j\neq k} \sqrt{\mathcal{E}_j}\,\rho_{jk}(0)\hat{b}_j\right] \tag{16.3–59}$$

Multistage interference cancellation (MIC) Multiuser detection based on MIC is a technique that employs multiple iterations in detecting the user bits and cancelling the interference. The method is easily described by means of an example.

EXAMPLE 16.3–1. TWO USERS AND SYNCHRONOUS TRANSMISSION. For the first stage of the detector, we may use the SIC detector or any of the suboptimum detectors. For example, suppose we use the decorrelating detector in the first stage.

First stage (decorrelating detector):

$$\begin{aligned}\hat{b}_1 &= \operatorname{sgn}(r_1 - \rho r_2)\\ \hat{b}_2 &= \operatorname{sgn}(r_2 - \rho r_1)\end{aligned}$$

Second stage:

$$\begin{aligned}\hat{\hat{b}}_1 &= \operatorname{sgn}\left(r_1 - \sqrt{\mathcal{E}_2}\hat{b}_2\rho\right)\\ \hat{\hat{b}}_2 &= \operatorname{sgn}\left(r_2 - \sqrt{\mathcal{E}_1}\hat{b}_1\rho\right)\end{aligned}$$

Third stage:

$$\begin{aligned}\hat{\hat{\hat{b}}}_1 &= \operatorname{gn}\left(r_1 - \sqrt{\mathcal{E}_2}\hat{\hat{b}}_2\rho\right)\\ \hat{\hat{\hat{b}}}_2 &= \operatorname{sgn}\left(r_2 - \sqrt{\mathcal{E}_1}\hat{\hat{b}}_1\rho\right)\end{aligned}$$

The computations may be terminated when there is no change in the decisions over two successive iterations.

Successive interference cancellation and multistage interference cancellation are two types of multiple access interference cancellation techniques that have received considerable attention by many researchers. For reference, we include the papers by Varanasi and Aazhang (1990), Patel and Holtzman (1994), Buehrer et al. (1996, 1999), and Divsalar et al. (1998).

We should indicate that the MIC is a suboptimum detector and does not converge to the jointly optimum multiuser detector defined above.

16.3–5 Other Types of Multiuser Detectors

Because of the widespread interest in the development of commercial CDMA communication systems, the design of multiuser detection algorithms continues to be a very active area of research. Our treatment in this chapter focused on the optimum MLSE algorithm, suboptimum linear (MMSE and decorrelating detection) algorithms, and non-linear successive interference cancellation algorithms based on hard decisions.

In addition to these relatively simple algorithms, a number of more complex algorithms have been described in the literature that are appropriate for time-dispersive channels which result in ISI. In addition, one may assume that knowledge of the signature waveforms of the other users is not available to a user receiver. Hence, a user receiver is confronted with both ISI and multiple access interference (MAI). In such a scenario, it is possible to design adaptive interference suppression algorithms that are akin to equalization algorithms previously described in Chapter 10.

Adapative algorithms for suppressing ISI and MAI in multiuser CDMA systems are described in the papers by Abdulrahman et al. (1994), Honig (1998), Miller (1995, 1996), Rapajic and Vucetic (1994), and Mitra and Poor (1995). In some cases, the adaptive algorithms are designed to converge without the use of any training symbols. Such algorithms are called *blind multiuser detection algorithms*. Examples of such blind algorithms are described in the papers by Honig et al. (1995), Madhow (1998), Wang and Poor (1998a, b), Bensley and Aazhang (1996) and the book by Wang and Poor (2004).

The use of multiple transmitting and/or receiving antennas in CDMA systems provides each user with the opportunity to employ spatial filtering in addition to temporal filtering to reduce ISI and MAI and combat signal fading. Blind multiuser detection algorithms for multiple antenna systems have been described by Wang and Poor (1999).

In general, the signals transmitted by the various users in a CDMA communication system are coded, either using a single level of coding or a concatenated code. Instead of separating the signal processing of the demodulator from the decoder, a better strategy is to use soft-information metrics from the decoder to enhance the suppression of the MAI and ISI at the demodulator. Thus, one can devise turbo-type iterative demodulation-decoding algorithms for suppressing MAI and ISI. Such algorithms for coded CDMA systems have been described in the papers by Reed et al. (1998), Moher (1998), Alexander et al. (1999), and Wang and Poor (1999).

16.3–6 Performance Characteristics of Detectors

The bit error probability is generally the desirable performance measure in multiuser communications. In evaluating the effect of multiuser interference on the performance of the detector for a single user, we may use as a benchmark the probability of a bit error for a single-user receiver in the absence of other users of the channel, which is

$$P_k(\gamma_k) = Q(\sqrt{2\gamma_k}) \tag{16.3–60}$$

where $\gamma_k = \mathcal{E}_k/N_0$, $\mathcal{E}_k$ is the signal energy per bit, and $\frac{1}{2}N_0$ is the power spectral density of the AWGN.

In the case of the optimum detector for either synchronous or asynchronous transmission, the probability of error is extremely difficult and tedious to evaluate. In this case, we may use Equation 16.3–60 as a lower bound and the performance of a suboptimum detector as an upper bound.

Let us consider, first, the suboptimum, conventional single-user detector. For synchronous transmission, the output of the correlator for the kth user is given by Equation 16.3–27. Therefore, the probability of error for the kth user, conditional on a sequence $\boldsymbol{b}_i$ of bits from other users, is

$$P_k(\boldsymbol{b}_i) = Q\left(\sqrt{2\left[\sqrt{\mathcal{E}_k} + \sum_{\substack{j=1\\ j\neq k}}^{K} \sqrt{\mathcal{E}_j}\, b_j(1)\rho_{jk}(0)\right]^2 \Big/ N_0}\right) \tag{16.3–61}$$

Then, the average probability of error is simply

$$P_k = \left(\tfrac{1}{2}\right)^{K-1} \sum_{i=1}^{2^{K-1}} P_k(\boldsymbol{b}_i) \tag{16.3–62}$$

The probability in Equation 16.3–62 will be dominated by the term that has the smallest argument in the Q function. The smallest argument will result in an SNR of

$$(\mathrm{SNR})_{\min} = \frac{1}{N_0}\left[\sqrt{\mathcal{E}_k} - \sum_{\substack{j=1\\ j\neq k}}^{K} \sqrt{\mathcal{E}_j}\,|\rho_{jk}(0)|\right]^2 \tag{16.3–63}$$

Therefore,

$$\left(\tfrac{1}{2}\right)^{K-1} Q\left(\sqrt{2(\mathrm{SNR})_{\min}}\right) < P_k < Q\left(\sqrt{2(\mathrm{SNR})_{\min}}\right) \tag{16.3–64}$$

A similar development can be used to obtain bounds on the performance for asynchronous transmission.

In the case of a decorrelating detector, the other-user interference is completely eliminated. Hence, the probability of error may be expressed as

$$P_k = Q\left(\sqrt{\mathcal{E}_k/\sigma_k^2}\right) \tag{16.3–65}$$

where σ_k^2 is the variance of the noise in the kth element of the estimate $\boldsymbol{b}^0$.

EXAMPLE 16.3–2. Consider the case of synchronous, two-user transmission, where $\boldsymbol{b}_2^0$ is given by Equation 16.3–41. Let us determine the probability of error.

The signal component for the first term in Equation 16.3–41 is $\sqrt{\mathcal{E}_1}$. The noise component is

$$n = \frac{n_1 - \rho n_2}{1 - \rho^2}$$

where ρ is the correlation between the two signature signals. The variance of this noise is

$$\begin{aligned} \sigma_1^2 &= \frac{E[(n_1 - \rho n_2)]^2}{(1-\rho^2)^2} \\ &= \frac{1}{1-\rho^2}\frac{N_0}{2} \end{aligned} \tag{16.3–66}$$

and

$$P_1 = Q\left(\sqrt{\frac{2\mathcal{E}_1}{N_0}(1-\rho^2)}\right) \tag{16.3–67}$$

A similar result is obtained for the performance of the second user. Therefore, the noise variance has increased by the factor $(1-\rho^2)^{-1}$. This noise enhancement is the price paid for the elimination of the multiuser interference by the decorrelating detector.

The error rate performance of the MMSE detector is similar to that for the decorrelating detector when the noise level is low. For example, from Equation 16.3–55, we observe that when N_0 is small relative to the diagonal elements of the signal correlation matrix $\boldsymbol{R}_N$,

$$\boldsymbol{b}^0 \approx \boldsymbol{R}_N^{-1}\boldsymbol{r} \tag{16.3–68}$$

which is the solution for the decorrelating detector. For low multiuser interference, the MMSE detector results in a smaller noise enhancement compared with the decorrelating detector, but has some residual bias resulting from the other users. Thus, the MMSE detector attempts to strike a balance between the residual interference and the noise enhancement.

An alternative to the error probability as a figure of merit that has been used to characterize the performance of a multiuser communication system is the ratio of SNRs with and without the presence of interference. In particular, Equation 16.3–60 gives the error probability of the kth user in the absence of other-user interference. In this case, the SNR is $\gamma_k = \mathcal{E}_k/N_0$. In the presence of multiuser interference, the user that transmits a signal with energy $\mathcal{E}_k$ will have an error probability P_k that exceeds $P_k(\gamma_k)$. The *effective SNR* γ_{ke} is defined as the SNR required to achieve the error probability

$$P_k = P_k(\gamma_{ke}) = Q(\sqrt{2\gamma_{ke}}) \tag{16.3–69}$$

The *efficiency* is defined as the ratio γ_{ke}/γ_k and represents the performance loss due to the multiuser interference. The desirable figure of merit is the *asymptotic efficiency*, defined as

$$\eta_k = \lim_{N_0 \to 0} \frac{\gamma_{ke}}{\gamma_k} \tag{16.3–70}$$

This figure of merit is often simpler to compute than the probability of error.

EXAMPLE 16.3–3. Consider the case of two symbol-synchronous users with signal energies $\mathcal{E}_1$ and $\mathcal{E}_2$. Let us determine the asymptotic efficiency of the conventional detector.

In this case, the probability of error is easily obtained from Equation 16.3–61 and Equation 16.3–62 as

$$P_1 = \tfrac{1}{2}Q\left(\sqrt{2(\sqrt{\mathcal{E}_1} + \rho\sqrt{\mathcal{E}_2})^2/N_0}\right) + \tfrac{1}{2}Q\left(\sqrt{2(\sqrt{\mathcal{E}_1} - \rho\sqrt{\mathcal{E}_2})^2/N_0}\right)$$

However, the asymptotic efficiency is much easier to compute. It follows from the definition of Equation 16.3–70 and from Equation 16.3–61 that

$$\eta_1 = \left[\max\left(0, 1 - \sqrt{\frac{\mathcal{E}_2}{\mathcal{E}_1}}|\rho|\right)\right]^2$$

A similar expression is obtained for η_2.

The asymptotic efficiency of the optimum and suboptimum detectors that we have described has been evaluated by Verdu (1986c), Lupas and Verdu (1989), and Xie et al. (1990b). Figure 16.3–3 illustrates the asymptotic efficiencies of these detectors when

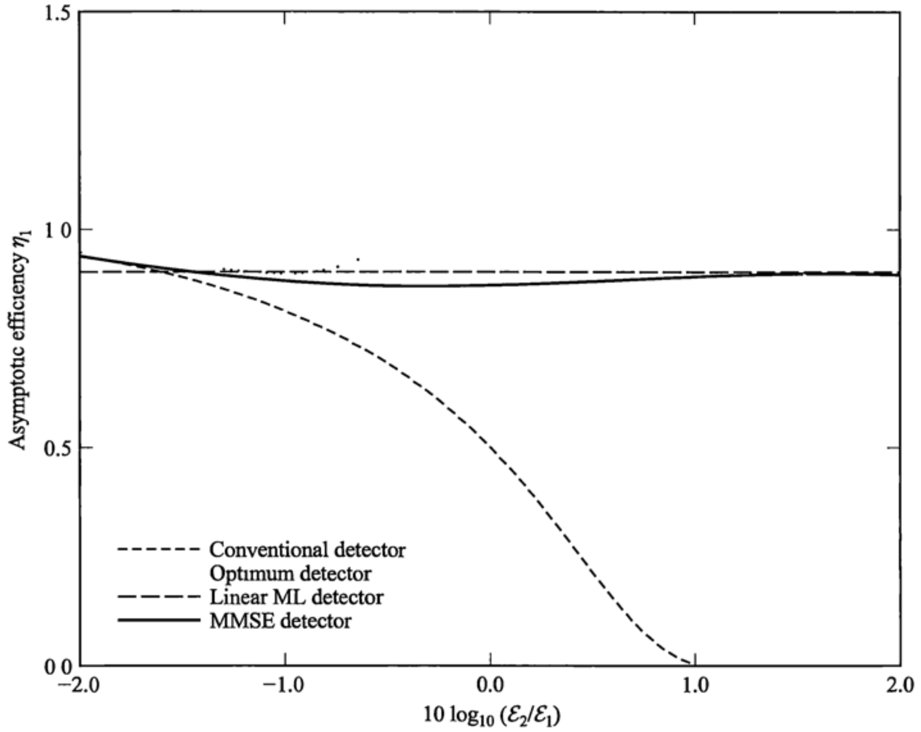


FIGURE 16.3–3
Asymptotic efficiencies of optimum (Viterbi) detector, conventional detector, MMSE detector, and linear ML detector in a two-user synchronous DS/SSMA system. [*From Xie* et al. *(1990 b),* © *IEEE.*]

$K = 2$ users are transmitting synchronously. These graphs show that when the interference is small ($\mathcal{E}_2 \to 0$), the asymptotic efficiencies of these detectors are relatively large (near unity) and comparable. As $\mathcal{E}_2$ increases, the asymptotic efficiency of the conventional single-user detector deteriorates rapidly. However, the other linear detectors perform relatively well compared with the optimum detector. Similar conclusions are reached by computing the error probabilities, but these computations are often more tedious.

16.4
MULTIUSER MIMO SYSTEMS FOR BROADCAST CHANNELS

In the previous section we treated the detection of signals transmitted simultaneously by multiple users to a common receiver. This scenario applies, for example, to the uplink of a cellular communication system in which the individual users transmit to a base station. We observed that the base station has the choice of selecting one of several multiuser detection methods to separate and recover the data transmitted by each of the multiple users.

In this section, we consider a broadcast scenario where data are transmitted simultaneously to multiple users from a common transmitting site. The transmitter is assumed to employ N_T antennas to transmit the data to K geographically distributed receivers, where $N_T \geq K$. Each user is assumed to have a receiver with one or more receiving antennas. This scenario applies, for example, to the downlink (broadcast mode) of a wireless local-area network (LAN) or a cellular communication system in which the channel is a MIMO channel. The distinguishing feature of this MIMO broadcast system is that the receivers are geographically distributed (point-to-multipoint transmission) and employ no coordination in processing the received signals. In contrast, the point-to-point MIMO systems that were treated in Chapter 15 exploited the availability of the signals from all the antennas in detecting the data.

In the MIMO broadcast scenario considered in this section, there are two possible approaches for dealing with the multiple-access interference (MAI) resulting from the simultaneous transmission to multiple users. One approach is to have each receiver employ interference mitigation in the recovery of its desired signal. In most cases, this approach is impractical because the users lack the processing capabilities and are constrained by the limited energy resources inherent in the use of battery power. The alternative approach is to employ interference mitigation techniques at the base station, which possesses significantly greater processing capabilities and energy resources. We adopt this more practical approach to interference mitigation for the MIMO broadcast channel.

MAI mitigation at the base station requires that the transmitter know the channel characteristics, typically the channel impulse response. This channel state information (CSI) may be obtained by channel measurements performed at each of the receivers by means of received pilot signals transmitted by the base station. Then the CSI must be transmitted to the base station for use in MAI mitigation. In some systems, the uplink and downlink channels are identical, e.g., the same frequency band is employed for both the uplink and downlink, but separate time slots are used for transmission. This transmission mode is called time-division duplex (TDD). In TDD operation, the pilot signals for channel measurement may be transmitted by each of the users in the uplink. In any case, we assume that the channel time variations are relatively slow so that a reliable estimate of the channel characteristics is available at the base station. In the treatment given in this section, we assume that the CSI at the transmitter is perfect.

The suppression of MAI by means of transmitter processing is usually called *signal precoding.* Although we will not include coded signal transmission in this discussion of MAI suppression, the addition of channel coding to achieve a rate near channel capacity is essential. In a paper entitled "Writing on Dirty Paper," Costa (1983) demonstrated that the capacity of an additive Gaussian noise channel further corrupted by additive interference that is known at the transmitter is the same as the capacity of the additive Gaussian noise channel without the additional interference. The analogy to writing on dirty paper is that if the writer (transmitter) knows where the dirt is located on the paper, the message can be written in a way that the reader (receiver) can recover the message without any knowledge of the location of the dirt. To elaborate, suppose the transmitter first selects a codeword $\mathbf{x}_1$, to be transmitted to receiver 1. Then the transmitter selects a codeword $\mathbf{x}_2$ to be transmitted to receiver 2, with knowledge of the codeword $\mathbf{x}_1$ to be sent to receiver 1. In such a case, the transmitter can presubtract $\mathbf{x}_1$ from $\mathbf{x}_2$, so that receiver 2 will receive $\mathbf{x}_2$ without interference. The signal precoding performed at the transmitter to suppress MAI is sometimes called *dirty paper precoding.*

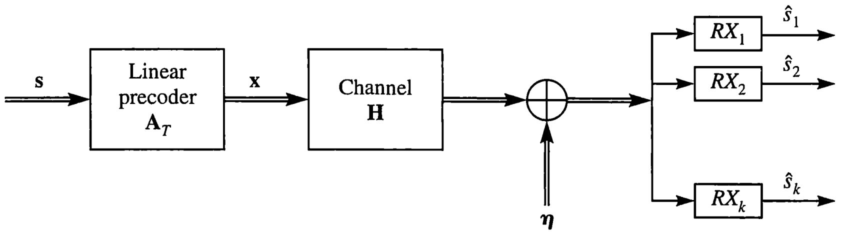


FIGURE 16.4–1
Model of MIMO broadcast system employing linear precoding.

Signal precoding at the transmitter may take one of several forms, depending on the criterion or the method used to perform the precoding. The simplest precoding methods are linear and are based on either the zero-forcing (ZF) criterion or the mean-square-error (MSE) criterion. Alternatively, there are nonlinear signal precoding methods that result in better system performance. We begin with a treatment of linear precoding and then we describe three nonlinear precoding methods.

16.4–1 Linear Precoding of the Transmitted Signals

For convenience and mathematical simplicity, we assume that each user has a single antenna and the number of receivers (users) is $K \leq N_T$. It is also convenient to assume that the channel is nondispersive. The communication system configuration is shown in Figure 16.4–1, where the precoding matrix is denoted as A_T. Hence, the received signal vector is

$$y = HA_T s + \eta \tag{16.4–1}$$

where H is a $K \times N_T$ matrix, A_T is an $N_T \times K$ matrix, s is a $K \times 1$ vector, and η is a $K \times 1$ Gaussian noise vector. The matrix that eliminates the MAI at each receiver is generally given by the Moore-Penrose pseudoinverse (see Appendix A)

$$H^+ = H^H(HH^H)^{-1} \tag{16.4–2}$$

Hence, the precoding matrix is

$$A_T = \alpha H^+ \tag{16.4–3}$$

where α is a scale factor that is selected to satisfy the total transmitted power allocation, i.e., $\|A_T s\|^2 = P$. Thus, the precoding matrix in Equation 16.4–3 allows the individual users to recover their desired symbols without any interference from the signals transmitted to the other users. We also observe that in the special case where $K = N_T$, $A_T = \alpha H^{-1}$. Furthermore, we note that when the symbols transmitted to the K users are selected from the same constellation, all users have the same SNR at their receivers and the corresponding data rates are also identical.

The sum capacity of the MIMO broadcast system that employs a channel inversion precoder has been investigated by Hochwald and Vishwanath (2005) and by Peel et al. (2005). It is shown in these references that the ergodic sum capacity with channel inversion, when $K = N_T \to \infty$, approaches a constant independent of K and N_T.

This result is in contrast to the achievable sum capacity of a MIMO system which, as we have observed, increases linearly as $\min(N_T, K)$. This poor performance resulting from channel inversion is attributed to the large disparity between the smallest and largest eigenvalues of the matrix $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$.

The effect of the ill-conditioning in the channel matrix $\boldsymbol{H}$ is also observed in the error rate performance of the MIMO broadcast system that employs channel inversion to suppress the MAI. This ill-conditioning requires an increase in transmit power to attain acceptable performance. The error rate performance is illustrated in the following example.

EXAMPLE 16.4–1. The broadcast system modeled by Equations 16.4–1 and 16.4–3 may be simulated on a computer. The channel matrix elements are complex-valued iid zero-mean Gaussian random variables with unit variance. The error rate performance of the zero-forcing precoder obtained via Monte Carlo simulation is illustrated in Figure 16.4–2 for $K = N_T = 4$, 6, and 10 for QPSK modulation. We observe that the error rate increases with an increase in the number of users. We attribute this deterioration in performance to the ill-conditioning of the channel matrix $\boldsymbol{H}$.

As we have observed, the major drawback with the zero-forcing solution is that when the channel matrix $\boldsymbol{H}$ is ill-conditioned (low gains or high attenuation in some of the transmitter-receiver links), the system performance is degraded, due to matrix inversion. If we relax the condition that the MAI be zero at all the receivers, the performance degradation can be reduced. This can be accomplished by using the linear MSE criterion in the design of the precoding matrix $\boldsymbol{A}_T$. Thus, we select $\boldsymbol{A}_T$ to minimize

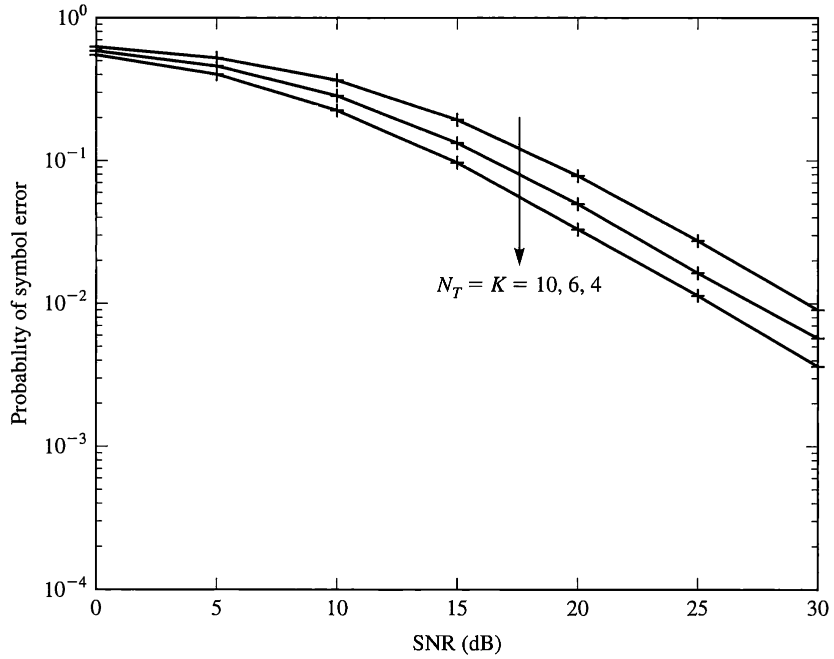


FIGURE 16.4–2

Performance of ZF linear precoding with $N_T = K = 4$, 6, 10. Performance improves as K decreases.

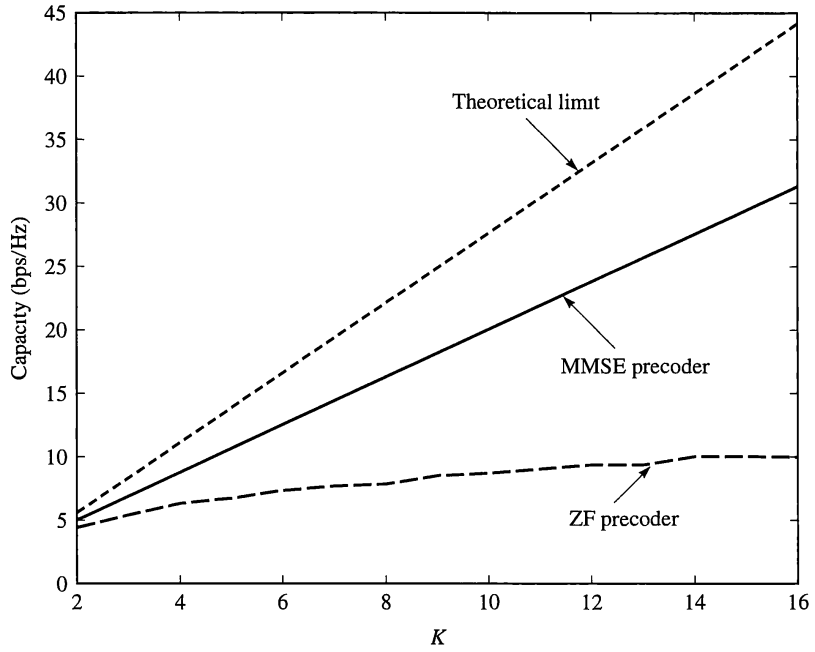


FIGURE 16.4–3
Comparison of the sum capacity for the linear precoder as a function of the number of users $K (K = N_T)$ for an SNR = 10 dB. [*From Peel et al. (2005).* © *IEEE.*]

the cost function

$$J(A_T, \alpha) = \arg\min_{\alpha, A_r} E \left\| \frac{1}{\alpha}(HA_{T}s + \eta) - s \right\|^2 \tag{16.4–4}$$

subject to the transmitted power allocation $\|A_T s\|^2 = P$, and where the expectation in Equation 16.4–4 is taken over the noise statistics and signal statistics. The solution to the MMSE criterion is the precoding matrix

$$A_T = \alpha H^H (HH^H + \beta I)^{-1} \tag{16.4–5}$$

where α is the scale factor that is selected to satisfy the power allocation and β is defined as a loading factor, which when selected as $\beta = K/P$ maximizes the signal-to-interference-plus-noise ratio (SINR) at the receiver [see Peel et al. (2005)].

Figure 16.4–3, taken from the paper by Peel et al. (2005), provides a comparison of the sum capacity for the two linear precoders based on the zero-forcing and the MMSE criteria. Also shown in this figure is the ergodic sum capacity of the MIMO channel when the channel characteristics are known at the transmitter. We observe that the sum capacity of the linear precoder designed on the basis of the MMSE criterion increases linearly with K, but it has a smaller slope than the theoretical limit.

The error rate performance of the MMSE linear precoder obtained by Monte Carlo simulation in a frequency-nonselective Rayleigh fading channel is illustrated in Figure 16.4–4 for $K = N_T = 4, 6,$ and 10. We observe that the error rate performance improves slightly as the number of users K increases.

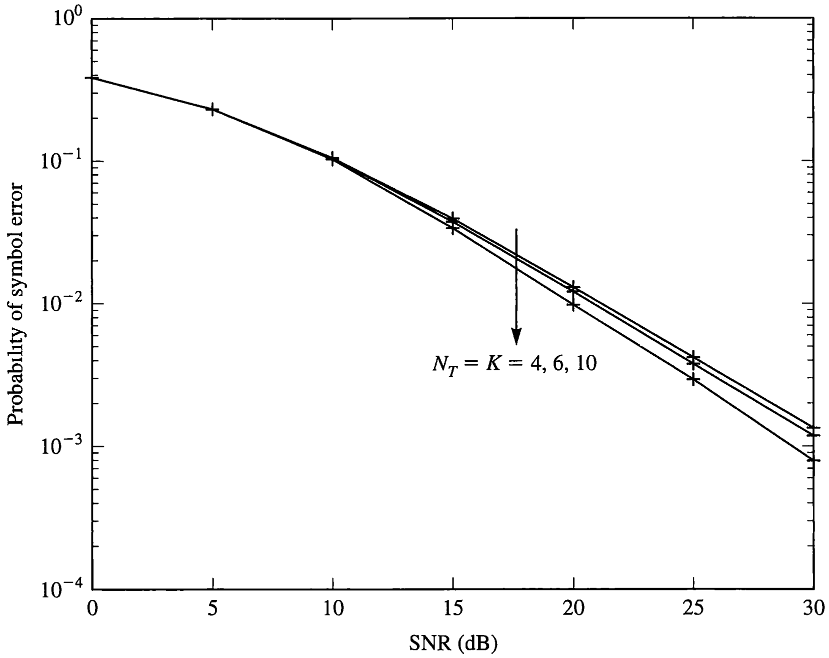


FIGURE 16.4–4
Performance of MMSE linear precoding with $N_T = K = 4, 6, 10$. Performance improves as K increases.

16.4–2 Nonlinear Precoding of the Transmitted Signals—The QR Decomposition

When the transmitter knows the interference caused on other users by the transmission of a signal to any particular user, the transmitter can design signals for each of the other users to cancel the MAI. The major problem with such an approach is to perform the interference cancellation without increasing the transmitter power. We encountered this same issue in our treatment of channel equalization based on decision-feedback equalization, where the feedback part of the equalizer was implemented at the transmitter (see Section 9.5–4). We recall that when the range of the difference between the desired symbol and the ISI exceeded the range of the desired transmitted symbol, the difference was reduced by subtracting an integer multiple of $2M$ for M-ary PAM, where $[-M, M)$ is the range of the desired transmitted signal. This same nonlinear precoding method, called Tomlinson-Harashima precoding, can be applied to the cancellation of the MAI in a MIMO broadcast communication system.

Figure 16.4–5 illustrates the precoding operations for the MIMO multiuser system. For a frequency-selective channel, the channel impulse response between the ith transmit antenna and the receive antenna of the kth user is modeled as

$$h_{ki}(t) = \sum_{l=0}^{L-1} h_{ki}^{(l)} \delta(t - lT) \tag{16.4–6}$$

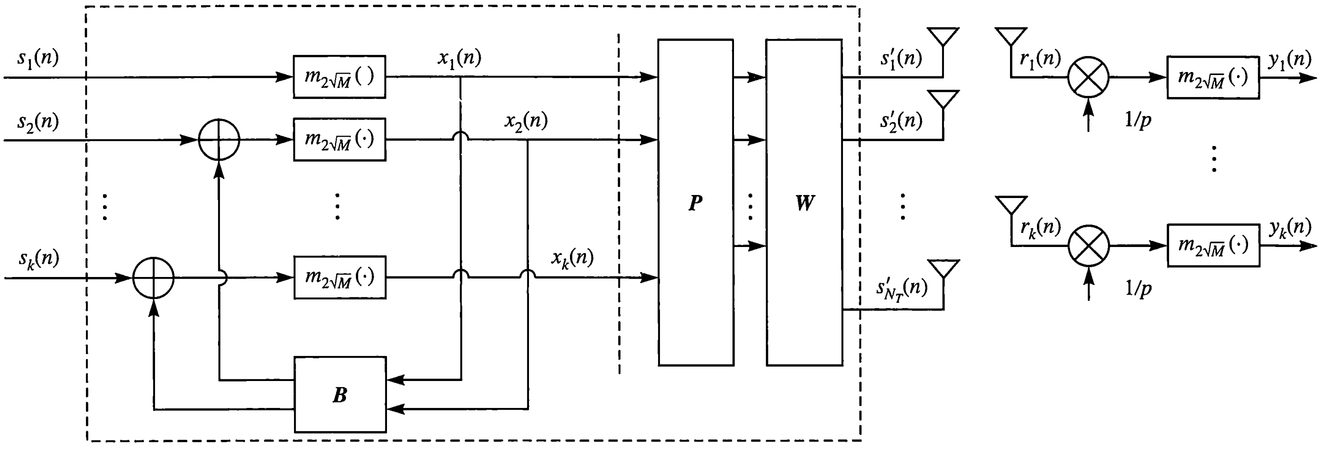


FIGURE 16.4–5
Tomlinson-Harashima precoding applied to a MIMO system.

where L is the number of multipath components in the channel response, T is the symbol duration, and $h_{ki}^{(l)}$ is the complex-valued channel coefficient for the lth path. The channel coefficients $\{h_{ki}^{(l)}\}$ are known at the transmitter and are realizations of iid zero-mean, circularly symmetric complex Gaussian random variables with variance

$$E\left[|h_{ki}^{(l)}|^2\right] = \frac{1}{L}, \qquad \forall k, i, \text{ and } l \tag{16.4–7}$$

It is convenient to arrange these channel coefficients for the lth path in a $K \times N_T$ matrix $\boldsymbol{H}^{(l)}$, where $[\boldsymbol{H}^{(l)}]_{ki} = h_{ki}^{(l)}$, $i = 1, 2, \ldots, N_T$, $k = 1, 2, \ldots, K$.

The MAI cancellation is facilitated by use of the QR decomposition of the channel matrix $\boldsymbol{H}^{(0)}$. Thus, we express $[\boldsymbol{H}^{(0)}]^H$ as

$$[\boldsymbol{H}^{(0)}]^H = \boldsymbol{Q}\boldsymbol{R} \tag{16.4–8}$$

where $\boldsymbol{Q}$ is an $N_T \times K$ matrix, such that $\boldsymbol{Q}\boldsymbol{Q}^H = \boldsymbol{I}$, and $\boldsymbol{R}$ is a $K \times K$ upper triangular matrix with diagonal elements $\{r_{ii}\}$. Based on this decomposition of $[\boldsymbol{H}^{(0)}]^H$, the signal to be transmitted is precoded with the matrix transformation

$$\boldsymbol{W} = \boldsymbol{Q}\boldsymbol{A} \tag{16.4–9}$$

where $\boldsymbol{A}$ is a $K \times K$ diagonal matrix with diagonal elements $1/r_{ii}, i = 1, 2, \ldots, K$. The $\{r_{ii}\}$ are real and positive [see Tulino and Verdu (2004)]. The matrix $\boldsymbol{P} = p\boldsymbol{I}$ is a diagonal $K \times K$ matrix that is used simply for scaling the power of the transmitted signal and results in equal SNR for all users. Therefore, we have an effective channel matrix of the form

$$\begin{aligned} \boldsymbol{H}^{(0)}\boldsymbol{W}\boldsymbol{P} &= [\boldsymbol{Q}\boldsymbol{R}]^H\boldsymbol{Q}\boldsymbol{A}\boldsymbol{P} \\ &= p\boldsymbol{R}^H\boldsymbol{A} \end{aligned} \tag{16.4–10}$$

We note that $\boldsymbol{R}^H\boldsymbol{A}$ is a $K \times K$ lower triangular matrix with unit diagonal elements. As a result, user k sees multiple access interference from users $1, 2, \ldots, k-1$. We also note that the effective channel matrix $\boldsymbol{H}^{(0)}\boldsymbol{W} = \boldsymbol{R}^{H}\boldsymbol{A}$ will have full rank K, provided that $N_T \geq K$.

By reducing this channel matrix to a lower triangular matrix, we can now subtract the interference at the transmitter that each user would normally observe at his or her respective receiver. Thus, when the channel adds the same interference to the transmitted signal, the received signal at each receiver will be free of interference. By taking advantage of the lower triangular matrix structure, successive interference cancellation is performed with the feedback filter defined by the matrix

$$\boldsymbol{B} = [\boldsymbol{I} - \boldsymbol{H}^{(0)}\boldsymbol{W}, -\boldsymbol{H}^{(1)}\boldsymbol{W}, -\boldsymbol{H}^{(2)}\boldsymbol{W}, \ldots, -\boldsymbol{H}^{(L-1)}\boldsymbol{W}] \tag{16.4–11}$$

where the matrix $(\boldsymbol{I} - \boldsymbol{H}^{(0)}\boldsymbol{W})$ is used to cancel the interference due to the other users that arises in the current symbol interval, and the terms $-\boldsymbol{H}^{(1)}\boldsymbol{W}, -\boldsymbol{H}^{(2)}\boldsymbol{W}, \ldots, -\boldsymbol{H}^{(L-1)}\boldsymbol{W}$ are used to cancel the interference due to previous symbols.

To ensure that the subtraction of the interference terms does not result in an increase of transmitter power, we use the modulo operator, as in Tomlinson–Harashima precoding, to limit the range of the signal to the boundaries of the signal constellation. Thus, the output of the modulo operators for the nth symbol vector, as shown in Figure 16.4–5, is (for square QAM constellations)

$$\begin{aligned} \boldsymbol{x}(n) &= \text{mod}\ _{2\sqrt{M}}[\boldsymbol{s}(n) + \boldsymbol{B}\hat{\boldsymbol{x}}(n)] \\ &= \boldsymbol{s}(n) + \boldsymbol{B}\hat{\boldsymbol{x}}(n) - 2\sqrt{M}\boldsymbol{z}_x(n) \end{aligned} \tag{16.4–12}$$

where the modulo operation is performed on each real and imaginary component of the vector $[\boldsymbol{s}(n) + \boldsymbol{B}\hat{\boldsymbol{x}}(n)]$, $\boldsymbol{x}(n)$ is the $K \times 1$ vector at the output of the modulo operator, $\boldsymbol{s}(n)$ is the $K \times 1$ data vector, $\hat{\boldsymbol{x}}(n)$ is defined as

$$\hat{\boldsymbol{x}}(n) = [\boldsymbol{x}(n)^t, \boldsymbol{x}(n-1)^t, \boldsymbol{x}(n-2)^t, \ldots, \boldsymbol{x}(n-(L-1))^t]^t \tag{16.4–13}$$

and $\boldsymbol{z}_x(n)$ is an $K \times 1$ vector with complex-valued components that take on integer values, determined by the constraint that the real and imaginary components of $\boldsymbol{x}(n)$ fall in the range of $[-\sqrt{M}, \sqrt{M})$. Therefore, the transmitted signal vector is expressed as

$$\begin{aligned} \boldsymbol{s}'(n) &= \boldsymbol{W}\boldsymbol{P}\boldsymbol{x}(n) \\ &= p\boldsymbol{W}\boldsymbol{x}(n) \end{aligned} \tag{16.4–14}$$

and the received signal vector is

$$\boldsymbol{r}(n) = p\sum_{i=0}^{L-1}\boldsymbol{H}^{(i)}\boldsymbol{W}\boldsymbol{x}(n-i) + \boldsymbol{\eta}(n) \tag{16.4–15}$$

Hence,

$$\boldsymbol{P}^{-1}\boldsymbol{r}(n) = \boldsymbol{x}(\boldsymbol{n}) + (\boldsymbol{H}^{(0)}\boldsymbol{W} - \boldsymbol{I})\boldsymbol{x}(n) + \sum_{i=1}^{L-1}\boldsymbol{H}^{(i)}\boldsymbol{W}\boldsymbol{X}(n-i) + \boldsymbol{\eta}'(n) \tag{16.4–16}$$

By substituting for $\boldsymbol{B}$ and $\boldsymbol{x}(n)$ in Equation 16.4–16, it follows that

$$\boldsymbol{P}^{-1}\boldsymbol{r}(n) = \boldsymbol{s}(n) + \boldsymbol{\eta}'(n) - 2\sqrt{M}\boldsymbol{z}_x(n) \qquad (16.4\text{–}17)$$

Consequently, the MAI and ISI canceled perfectly, resulting in the test statistics for the nth symbol vector as

$$\boldsymbol{y}(n) = \text{mod}_{2\sqrt{M}}\left[\frac{1}{p}\boldsymbol{r}(n)\right] \qquad (16.4\text{–}18)$$

Optimum Ordering of the Decentralized Receivers

The ordering of the K decentralized receivers affects the construction of the $K \times N_T$ channel matrix $\boldsymbol{H}^{(0)}$. There are $K!$ possible column permutations of $[\boldsymbol{H}^{(0)}]^H$, and hence there is one QR decomposition associated with each permutation. In turn, there are $K!$ transformation matrices $\boldsymbol{W} = \boldsymbol{Q}\boldsymbol{A}$, each of which requires a different transmit power. To minimize the total transmit power, it is necessary to search over all the column permutations of $[\boldsymbol{H}^{(0)}]^H$. Such an exhaustive search procedure is computationally time-consuming, except for a small number of users. Foschini et al. (1999) have described methods for simplifying the search for the optimum ordering.

The error rate performance of the QR decomposition method described above has been evaluated by Amihood et al. (2006, 2007). Figure 16.4–6 illustrates the symbol error probability as a function of the SNR (total transmitted signal power over all antennas divided by N_0) for QPSK modulation, $L = 1, 2$ and $N_T = K = 2$. The

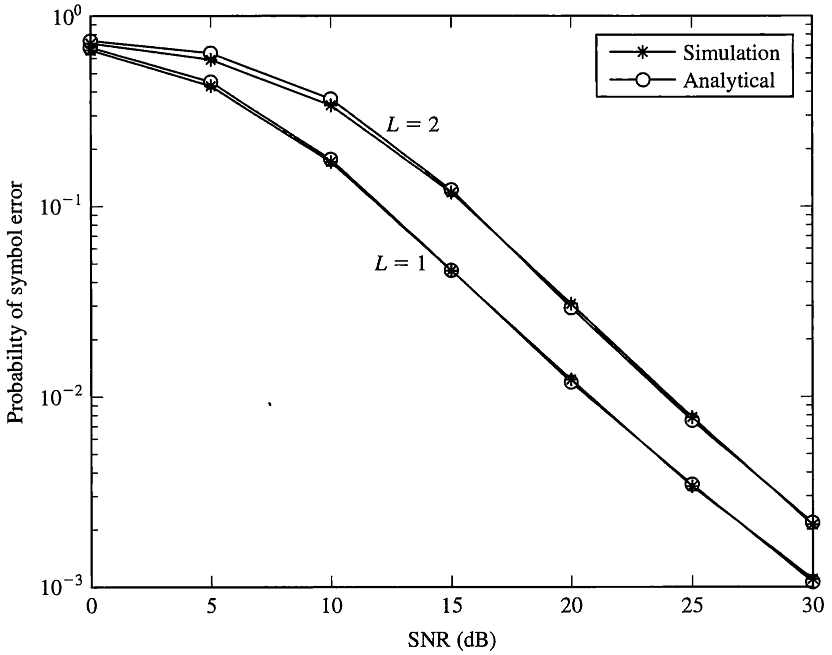


FIGURE 16.4–6
Performance of optimal QR decomposition with $N_T = K = 2$ and $L = 1$ and 2.

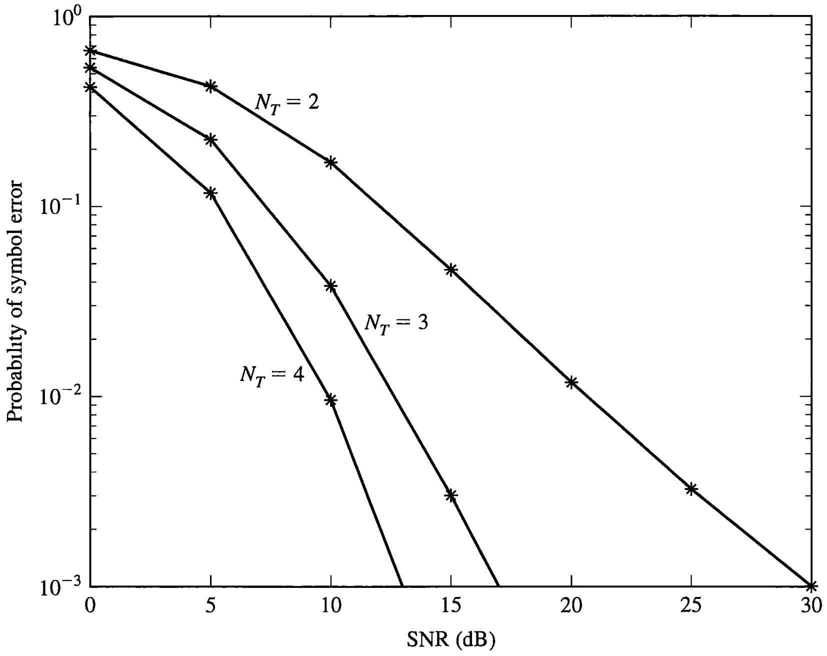


FIGURE 16.4–7
Performance of optimal ordered QR decomposition with $K = 2$, $L = 1$ and $N_T = 2, 3$, and 4.

Monte Carlo simulation results are also illustrated. The simulation results are obtained by transmitting 1000 data symbols over each of 10,000 channel realizations.

Figure 16.4–7 shows the symbol error rate performance for QPSK with $L = 1$ (flat fading), $K = 2$, and $N_T = 2, 3, 4$. We observe that the system performance improves with an increase in the number of transmit antennas, which reflects the benefit of spatial diversity.

Figure 16.4–8 shows a comparison of the error rate performance of the linear ZF and MMSE precoding methods with the QR decomposition method for QPSK modulation with $L = 1$ and $K = N_T = 4$. Figure 16.4–9 shows a similar comparison for $K = N_T = 6$. We observe that the performance of the QR decomposition method is better than that of the linear precoders at high SNRs but poorer at low SNRs. However, the improvement in performance of the QR decomposition method at high SNRs should be weighed against the significantly higher computational complexity compared with the linear MMSE precoder.

16.4–3 Nonlinear Vector Precoding

The QR decomposition method described in Section 16.4–2 is one of several nonlinear precoding techniques described in the literature for suppressing MAI in MIMO broadcast communication systems. These methods may be generally described as vector precoding techniques.

Hochwald et al. (2005) have proposed and evaluated the performance of a vector precoding technique in which the data vector to be transmitted to the K users is modified by the addition of a precoding vector with integer elements. In particular, let us consider

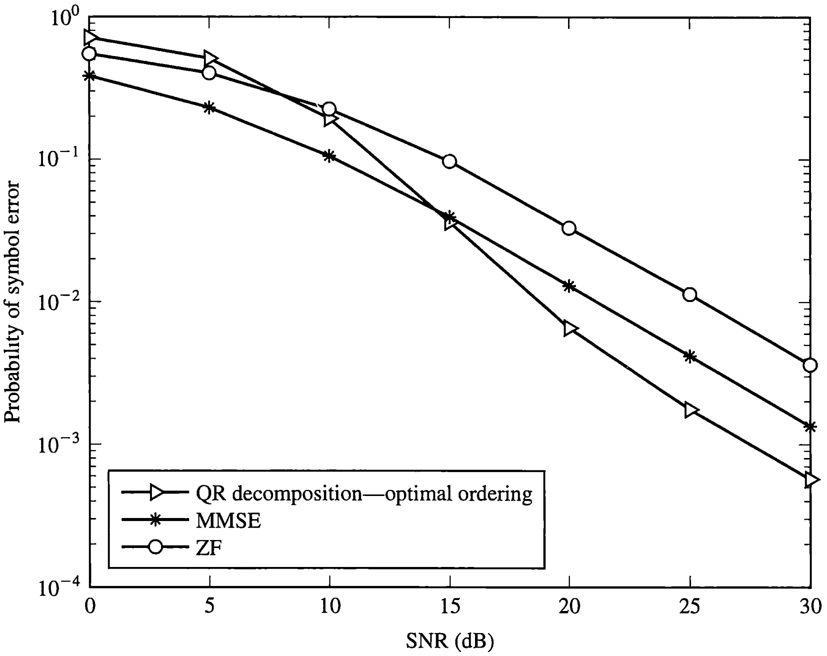


FIGURE 16.4–8
Comparison of the QR decomposition and the linear precoders with $N_T = K = 4$.

a modification of the linear zero-forcing precoder in which each element of the data vector $\boldsymbol{s}$ is offset by some judiciously selected integer, as illustrated in Figure 16.4–10. Thus, the offset data vector becomes

$$\boldsymbol{s}' = \boldsymbol{s} + \tau \hat{\boldsymbol{p}} \tag{16.4–19}$$

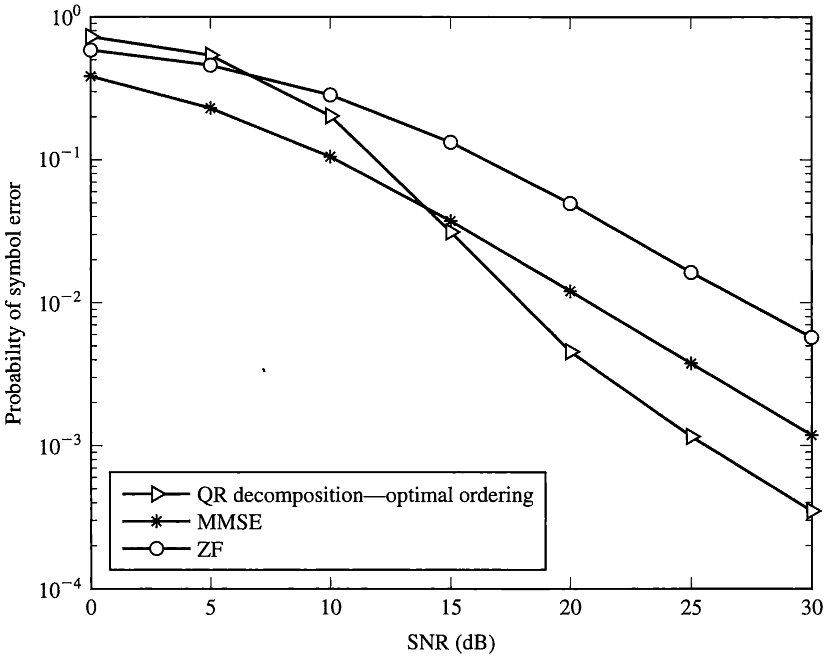


FIGURE 16.4–9
Comparison of the QR decomposition and the linear precoders with $N_T = K = 6$.

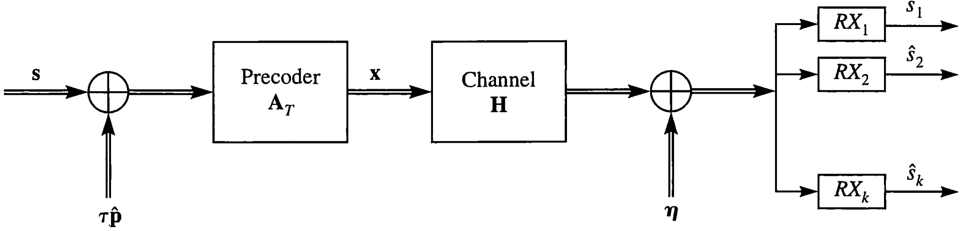


FIGURE 16.4–10
Model of MIMO broadcast system employing vector precoding.

where τ is a real positive number and $\hat{\boldsymbol{p}}$ is a K-dimensional vector with complex-valued elements, where the real and imaginary components are integers. Hence, for $N_T = K$, the transmitted signal vector is

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{A}_T(\boldsymbol{s} + \tau\hat{\boldsymbol{p}}) \\ &= \alpha \boldsymbol{H}^{-1}(\boldsymbol{s} + \tau\hat{\boldsymbol{p}}) \end{aligned} \tag{16.4–20}$$

The offset vector $\hat{\boldsymbol{p}}$ is chosen to minimize the power in the transmitted signal, i.e.,

$$\hat{\boldsymbol{p}} = \arg\min_{\boldsymbol{p}} \parallel \alpha \boldsymbol{H}^{-1}(\boldsymbol{s} + \tau\boldsymbol{p}) \parallel^2 \tag{16.4–21}$$

Hence, the vector perturbation method jointly optimizes the perturbation vector for the signals that are transmitted to all the receivers. Algorithms for solving this least-squares K-dimensional integer-lattice problem are given in the paper by Hochwald et al. (2005).

It is demonstrated in Hochwald et al. (2005) that the optimization of the perturbation vector $\boldsymbol{p}$ results in an offset data vector $\boldsymbol{s}'$ that, on average, is oriented toward each eigenvalue of $(\boldsymbol{H}\boldsymbol{H}^H)^{-1}$ in inverse proportion to the eigenvalue. This vector precoding method generally yields better error rate performance than the QR decomposition method, described in the previous section, that employs scalar Tomlinson–Harashima precoding.

The perturbation vector $\hat{\boldsymbol{p}}$ is not known to the receivers. However, by constraining the elements of $\hat{\boldsymbol{p}}$ to be integers, the receivers may use the modulo operation, as in Tomlinson–Harashima precoding, to recover the data components. The scalar τ is selected large enough that each receiver applies the modulo function to the real and imaginary components of each element of the received vector $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\eta}$ to recover the corresponding element of the data vector $\boldsymbol{s}$. It is desirable to choose τ so that it results in a symmetric decoding region around the real and imaginary components of every signal constellation symbol. The choice of τ that accomplishes this desired goal is

$$\tau = 2|s_k|_{\max} + \Delta \tag{16.4–22}$$

where $|s_k|_{\max}$ is the signal constellation symbol having the largest magnitude and Δ is the distance between adjacent constellation symbols.

The vector perturbation technique may also be applied to the linear precoder based on the MMSE criterion. In this case, the transmitted vector is

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{A}_T(\boldsymbol{s} + \tau \hat{\boldsymbol{p}}) \\ &= \alpha \boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H + \beta \boldsymbol{I})^{-1}(\boldsymbol{s} + \tau \hat{\boldsymbol{p}}) \end{aligned} \tag{16.4–23}$$

where $\hat{\boldsymbol{p}}$ is selected to minimize the power of the transmitted signal, i.e.,

$$\hat{\boldsymbol{p}} = \arg\min_{\boldsymbol{p}} \|\alpha \boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H + \beta \boldsymbol{I})^{-1}(\boldsymbol{s} + \tau \boldsymbol{p})\|^2 \tag{16.4–24}$$

where α is selected to satisfy the transmitted power allocation constraint, β is selected to maximize the signal-to-interference-plus-noise ratio, and τ is selected as described previously to result in a symmetric decoding region around the real and imaginary components of every signal constellation symbol. Hence the received signal vector is

$$\boldsymbol{r} = \alpha \boldsymbol{H}\boldsymbol{H}^H(\boldsymbol{H}\boldsymbol{H}^H + \beta \boldsymbol{I})^{-1}(\boldsymbol{s} + \tau \hat{\boldsymbol{p}}) + \boldsymbol{\eta} \tag{16.4–25}$$

The mth user assumes that its received signal has the form

$$r_m = \alpha(s_m + \tau p_m) + \eta'_m \tag{16.4–26}$$

where η'_m includes the additive channel noise and the MAI from other users due to the nonzero scale factor β. Since each user knows α and τ, the mth user performs the modulo operation on r_m to remove p_m and passes the result to its decoder. It is demonstrated in Hochwald et al. (2005) that the performance of this vector perturbation scheme is significantly better than the linear MMSE precoder described in Section 16.4–1.

16.4–4 Lattice Reduction Technique for Precoding

Lattice constellations are quite common in designing signal sets for communication systems. We have studied the main properties of lattices and lattice-based constellations in Section 4.7. Lattice precoding is a technique similar to the Tomlinson–Harashima precoding that can be used with channels with known interference at the transmitter.

We consider the MIMO broadcast channel model with N_T transmit antennas at the base station and K receivers each with a single antenna. We also assume $K \leq N_T$. The input-output relation for the channel is written as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{\eta} \tag{16.4–27}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the transmitted and received signals with N_T and K components, respectively, $\boldsymbol{\eta}$ is a vector of iid random variables each drawn according to $\mathcal{CN}(0, N_0)$, and $\boldsymbol{H}$ is a $K \times N_T$ matrix of complex channel coefficients. As previously stated, the matrix $\boldsymbol{H}$ is assumed to be perfectly known at the transmitter.

The original lattice reduction techniques were developed for real lattices and in order to employ them it is convenient to introduce a real equivalent of the communication system under study. Equation 16.4–27 is equivalent to the following form in which all quantities are real

$$\begin{bmatrix}\mathrm{Re}(\boldsymbol{y})\\ \mathrm{Im}(\boldsymbol{y})\end{bmatrix}\begin{bmatrix}\mathrm{Re}(\boldsymbol{H}) & -\mathrm{Im}(\boldsymbol{H})\\ \mathrm{Im}(\boldsymbol{H}) & \mathrm{Re}(\boldsymbol{H})\end{bmatrix}\begin{bmatrix}\mathrm{Re}(\boldsymbol{x})\\ \mathrm{Im}(\boldsymbol{x})\end{bmatrix}+\begin{bmatrix}\mathrm{Re}(\boldsymbol{\eta})\\ \mathrm{Im}(\boldsymbol{\eta})\end{bmatrix} \tag{16.4–28}$$

This equation can be written as

$$\boldsymbol{y}_r = \boldsymbol{H}_r\boldsymbol{x}_r + \boldsymbol{\eta}_r \tag{16.4–29}$$

The vector of data symbols intended for the K receivers is denoted by $\boldsymbol{s}$, which is a K-dimensional vector with components in an M-ary QAM constellation which is defined as a set of lattice points with a given boundary.

We have seen different types of precoding in the previous sections, among them zero-forcing precoding matrix of the form $\boldsymbol{A}_{Tr} = \alpha\boldsymbol{H}_r^+ = \alpha\boldsymbol{H}_r^H(\boldsymbol{H}_r\boldsymbol{H}_r^H)^{-1}$ resulting in

$$\boldsymbol{x}_r = \boldsymbol{A}_{Tr}\boldsymbol{s}_r = \alpha\boldsymbol{H}_r^H\left(\boldsymbol{H}_r\boldsymbol{H}_r^H\right)^{-1}\boldsymbol{s}_r \tag{16.4–30}$$

and MMSE precoding matrix of the form $\boldsymbol{A}_{Tr} = \alpha\boldsymbol{H}_r^H\left(\boldsymbol{H}_r\boldsymbol{H}_r^H + \beta\boldsymbol{I}\right)^{-1}$ resulting in

$$\boldsymbol{x}_r = \boldsymbol{A}_{Tr}\boldsymbol{s}_r = \alpha\boldsymbol{H}_r^H\left(\boldsymbol{H}_r\boldsymbol{H}_r^H + \beta\boldsymbol{I}\right)^{-1}\boldsymbol{s}_r \tag{16.4–31}$$

as examples of linear precoding, and Tomlinson–Harashima which uses modulo arithmetic at the transmitter and requires a modulo operation at the receiver before quantizing to the M-ary QAM constellation. This nonlinear precoding technique is based on the QR decomposition of $\boldsymbol{H}_r$ and successive cancellation whose performance can be improved by optimal ordering of the subchannels using the algorithm described by Foschini et al. (1999).

The perturbation method of Section 16.4–3 can also be expressed in terms of the real equivalent matrix representation of Equation 16.4–29 as

$$\begin{aligned}\boldsymbol{x}_r &= \boldsymbol{A}_{Tr}(\boldsymbol{s}_r + \hat{\boldsymbol{p}})\\ \hat{\boldsymbol{p}} &= \arg\min_{\boldsymbol{p}'\in\alpha\mathbb{Z}^{2K}} \|\boldsymbol{A}_{Tr}(\boldsymbol{s}_r + \boldsymbol{p}')\|^2\end{aligned} \tag{16.4–32}$$

where $\mathbb{Z}^{2K}$ is the $2K$-dimensional integer lattice and α is the scalar $(2\sqrt{M})$ in the Tomlinson–Harashima modulo operation. The optimization of $\boldsymbol{p}$ in Equation 16.4–32 can be interpreted as finding the closest point in the lattice $\alpha\boldsymbol{A}_{Tr}\mathbb{Z}^{2K}$ to $-\boldsymbol{A}_{Tr}\boldsymbol{s}_r$, which can be accomplished using the Voronoi regions of the lattice.

As studied in Section 4.7, a lattice can be expressed in terms of its generator matrix $\boldsymbol{G}$ whose rows denote a basis for the lattice; i.e., all lattice points can be written as a linear combination of the rows of $\boldsymbol{G}$ with integer coefficients. Any lattice Λ can have many generator matrices and many bases for representation of lattice points. In particular, if $\boldsymbol{F}$ is a square matrix with integer entries such that $\det\boldsymbol{F} = \pm 1$, then $\boldsymbol{F}^{-1}$ exists and its entries are all integers. Then $\boldsymbol{G}' = \boldsymbol{F}\boldsymbol{G}$ is a generator of lattice Λ. The new generator matrix $\boldsymbol{G}'$ defines a new basis for the lattice Λ. A desirable property of the modified lattice basis is that it be an orthogonal or close-to-orthogonal basis with the lowest basis vector norms. The process of finding such a basis for a lattice is called *lattice reduction*. Although lattice reduction in high dimensions is an NP-hard problem, a polynomial-time suboptimal lattice reduction method due to Lenstra, Lenstra, and Lovász, known as the LLL algorithm for lattice reduction, exists that in most cases gives very good results (Lenstra et al. (1982)).

Since we are looking for $\boldsymbol{p}$ in lattice $\alpha \boldsymbol{A}_{Tr}\mathbb{Z}^{2K}$ that is closest to $-\boldsymbol{A}_{Tr}\boldsymbol{s}_r$, we can apply the LLL algorithm and write

$$\boldsymbol{A}_{Tr} = \boldsymbol{W}_r \boldsymbol{F}_r \qquad (16.4\text{–}33)$$

where $\boldsymbol{W}_r$ is a real-valued $2N_T \times 2K$ matrix, representing the transformed close-to-orthogonal basis and $\boldsymbol{F}_r$ is the integer-valued matrix with $\det \boldsymbol{F}_r = \pm 1$ that represents the transformation. A benefit of a close-to-orthogonal basis with low basis vector norm is that when linear interference mitigation techniques are applied to this bases, noise enhancement effects are lower.

In Figure 16.4–11 the left diagram shows the lattice corresponding to $\alpha \boldsymbol{A}_{Tr}\mathbb{Z}^2$ with its Voronoi regions representing minimum-distance solutions of Equation 16.4–32. The original basis for this lattice is denoted by the dashed arrows. Applying LLL to this lattice results in the reduced basis denoted by solid arrows which are closer to an orthogonal basis compared to the original basis. If we use the original basis for linear equalization, we obtain the figure shown in the middle in which the dashed arrows are orthonormal. However, the integer grid shown with dashed boundaries does not match the modified Voronoi regions. In fact, large white areas that correspond to the mismatch between the two regions indicate the inefficiency of this approach. In the rightmost figure, the result of applying linear equalization to the reduced basis is shown. As seen here, there is good overlap between the modified Voronoi regions and the integer grid, indicating the efficiency of this method.

The lattice reduction method has also been applied directly to lattices in complex dimensions using a complex version of the LLL algorithm as described by Gan and Mow (2005). In this case the lattice is described by n linear independent complex row vectors $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_n$ of length n that constitute a basis for the lattice. All lattice points

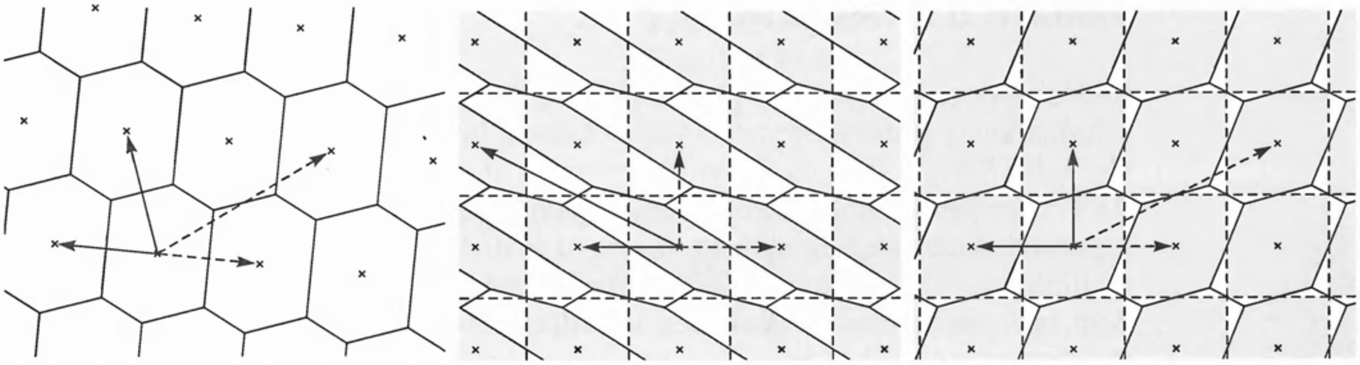


FIGURE 16.4–11
Left: Lattice $A\boldsymbol{H}_r^{+}\mathbb{Z}^2$ and its Voronoi regions with original basis (dashed) and modified basis (solid). *Middle*: Linear equalization applied to the original basis. *Right*: Linear equalization applied to the modified basis. [*From Windpassinger et al. (2004), copyright IEEE.*]

can be written as

$$x = \sum_{i=1}^{n} c_i g_i \tag{16.4–34}$$

where c_i's are complex numbers with integer real and imaginary parts and matrix $\boldsymbol{G}$ whose rows are $\boldsymbol{g}_i$'s is the generator of the lattice. Similar to real lattices, if $\boldsymbol{G}' = \boldsymbol{GF}$ and $\boldsymbol{F}$ is a square matrix with complex entries with integer real and imaginary parts such that $\det \boldsymbol{F} = \pm 1$ or $\det \boldsymbol{F} = \pm j$. Then $\boldsymbol{G}'$ is also a basis for the lattice generated by $\boldsymbol{G}$. The complex LLL reduction is of the form $\boldsymbol{A}_T = \boldsymbol{WF}$ where $\boldsymbol{W}$ represents the close-to-orthogonal reduced basis.

Depending on the approach selected, $\boldsymbol{A}_T$ can have different forms. For the zero-forcing approach $\boldsymbol{A}_T = \alpha \boldsymbol{H}^+ = \alpha \boldsymbol{H}^H(\boldsymbol{HH}^H)^{-1}$ and for the MMSE approach $\boldsymbol{A}_T = \alpha \boldsymbol{H}^H(\boldsymbol{HH}^H + \beta \boldsymbol{I})^{-1}$. For the perturbation method which employs Voronoi regions to find the closest lattice point, the approximate offset vector is given by

$$\boldsymbol{p}_{\text{approx}} = -\boldsymbol{F}^{-1} Q(\boldsymbol{Fs}) \tag{16.4–35}$$

where $Q(\cdot)$ denotes the componentwise rounding of the K-dimensional vector to the scaled complex integer lattice.

The lattice reduction technique studied by Windpassinger et al. (2004) indicates the effectiveness of this method in improving the performance through increasing the diversity gain. In fact the order of signal diversity achieved by the lattice reduction technique is comparable to the signal diversity obtained by the maximum-likelihood detection, but this signal diversity in the lattice reduction technique is obtained at a much lower complexity. The interested reader is referred to Yao and Wornell (2002), Fischer and Windpassinger (2003), and Windpassinger et al. (2004) for details.

16.5
RANDOM ACCESS METHODS

In this section, we consider a multiuser communication system in which users transmit information in packets over a common channel. In contrast to the CDMA method described in Section 16.3, the information signals of the users are not spread in frequency. As a consequence, simultaneous transmission of signals from multiple users cannot be separated at the receiver, without the use of spatial filtering which can be achieved by multiple receiving antennas. The access methods described below are basically random, because packets are generated according to some statistical model. Users access the channel when they have one or more packets to transmit. When more than one user attempts to transmit packets simultaneously, the packets overlap in time, i.e., they collide, and, hence, a conflict results, which must be resolved by devising some channel protocol for retransmission of the packets. Below, we describe several random access channel protocols that resolve conflicts in packet transmission.

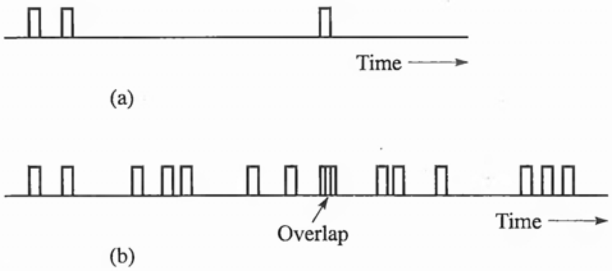


FIGURE 16.5–1
Random access packet transmission:
(**a**) packets from a typical user;
(**b**) packets from several users.

16.5–1 ALOHA Systems and Protocols

Suppose that a random access scheme is employed where each user transmits a packet as soon as it is generated. When a packet is transmitted by a user and no other user transmits a packet for the duration of the time interval, then the packet is considered successfully transmitted. However, if one or more of the other users transmits a packet that overlaps in time with the packet from the first user, a collision occurs and the transmission is unsuccessful. Figure 16.5–1 illustrates this scenario. If the users know when their packets are transmitted successfully and when they have collided with other packets, it is possible to devise a scheme, which we may call a *channel access protocol*, for retransmission of collided packets.

Feedback to the users regarding the successful or unsuccessful transmission of packets is necessary and can be provided in a number of ways. In a radio broadcast system, such as one that employs a satellite relay as depicted in Figure 16.5–2, the packets are broadcast to all the users on the downlink. Hence, all the transmitters can monitor their transmissions and, thus, obtain the following ternary information: no packet was transmitted, or a packet was transmitted successfully, or a collision occurred. This type of feedback to the transmitters is generally denoted as $(0, 1, c)$ feedback. In systems that employ wireline or filter-optic channels, the receiver may transmit the feedback signal on a separate channel.

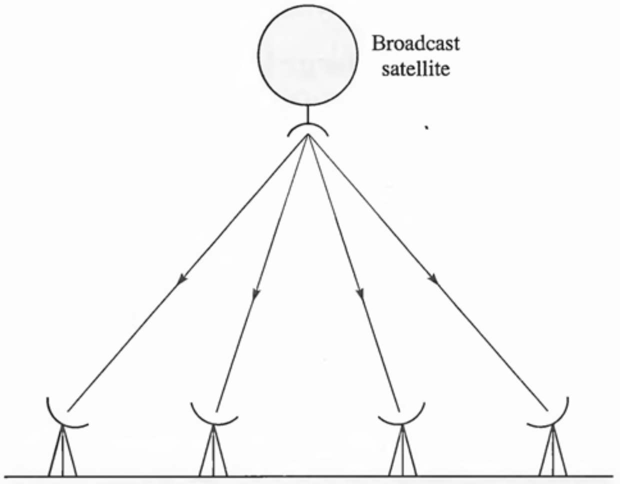


FIGURE 16.5–2
Broadcast system.

The ALOHA system devised by Abramson (1970, 1977) and others at the University of Hawaii employs a satellite repeater that broadcasts the packets received from the various users who access the satellite. In this case, all the users can monitor the satellite transmissions and, thus, establish whether or not their packets have been transmitted successfully.

There are basically two types of ALOHA systems: *synchronized or slotted* and *unsynchronized or unslotted.* In an unslotted ALOHA system, a user may begin transmitting a packet at any arbitrary time. In a slotted ALOHA, the packets are transmitted in time slots that have specified beginning and ending times.

We assume that the start time of packets that are transmitted is a Poisson point process having an average rate of λ packets/s. Let T_p denote the time duration of a packet. Then, the normalized channel traffic G, also called the *offered channel traffic*, is defined as

$$G = \lambda T_p \tag{16.5–1}$$

There are many channel access protocols that can be used to handle collisions. Let us consider the one due to Abramson (1973). In Abramson's protocol, packets that have collided are retransmitted with some delay τ, where τ is randomly selected according to the PDF

$$p(\tau) = \alpha e^{-\alpha\tau} \tag{16.5–2}$$

where α is a design parameter. The random delay τ is added to the time of the initial transmission and the packet is retransmitted at the new time. If a collision occurs again, a new value of τ is randomly selected and the packet is retransmitted with a new delay from the time of the second transmission. This process is continued until the packet is transmitted successfully. The design parameter α determines the average delay between retransmissions. The smaller the value of α, the longer the delay between retransmissions.

Now, let λ', where $\lambda' < \lambda$, be the rate at which packets are transmitted successfully. Then, the normalized channel throughput is

$$S = \lambda' T_p \tag{16.5–3}$$

We can relate the channel throughput S to the offered channel traffic G by making use of the assumed start time distribution. The probability that a packet will not overlap a given packet is simply the probability that no packet begins T_p seconds before or T_p seconds after the start time of the transmitted packet. Since the start time of all packets is Poisson-distributed, the probability that a packet will not overlap is $\exp(-2\lambda T_p) = \exp(-2G)$. Therefore,

$$S = Ge^{-2G} \tag{16.5–4}$$

This relationship is plotted in Figure 16.5–3. We observe that the maximum throughput is $S_{\max} = 1/2e = 0.184$ packets per slot, which occurs at $G = \frac{1}{2}$. When $G > \frac{1}{2}$, the throughput S decreases. The above development illustrates that an unsynchronized or unslotted random access method has a relatively small throughput and is inefficient.

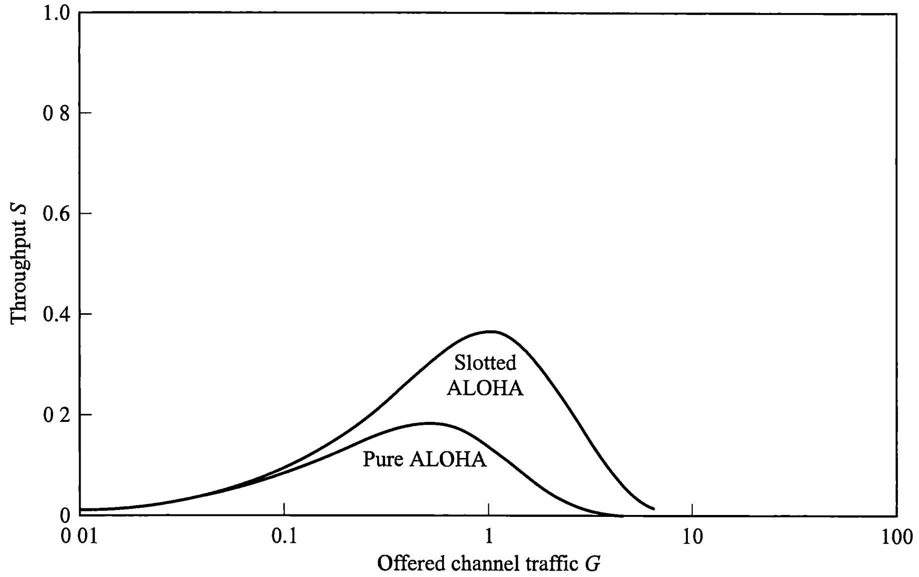


FIGURE 16.5-3
Throughput in ALOHA systems.

Throughput for slotted ALOHA To determine the throughput in a slotted ALOHA system, let G_i be the probability that the ith user will transmit a packet in some slot. If all the K users operate independently and there is no statistical dependence between the transmission of the user's packet in the current slot and the transmission of the user's packet in previous time slots, the total (normalized) offered channel traffic is

$$G = \sum_{i=1}^{K} G_i \tag{16.5-5}$$

Note that, in this case, G may be greater than unity.

Now, let $S_i \leq G_i$ be the probability that a packet transmitted in a time slot is received without a collision. Then, the normalized channel throughput is

$$S = \sum_{i=1}^{K} S_i \tag{16.5-6}$$

The probability that a packet from the ith user will not have a collision with another packet is

$$Q_i = \prod_{\substack{j=1 \\ j \neq i}}^{K} (1 - G_j) \tag{16.5-7}$$

Therefore,

$$S_i = G_i Q_i \tag{16.5-8}$$

A simple expression for the channel throughput is obtained by considering K identical users. Then,

$$S_i = \frac{S}{K}, \qquad G_i = \frac{G}{K}$$

and

$$S = G\left(1 - \frac{G}{K}\right)^{K-1} \tag{16.5–9}$$

Then, if we let $K \to \infty$, we obtain the throughput

$$S = Ge^{-G} \tag{16.5–10}$$

This result is also plotted in Figure 16.5–3. We observe that S reaches a maximum throughput of $S_{\max} = 1/e = 0.368$ packets per slot at $G = 1$, which is twice the throughput of the unslotted ALOHA system.

The performance of the slotted ALOHA system given above is based on Abramson's protocol for handling collisions. A higher throughput is possible by devising a better protocol.

A basic weakness in Abramson's protocol is that it does not take into account the information on the amount of traffic on the channel that is available from observation of the collisions that occur. An improvement in throughput of the slotted ALOHA system can be obtained by using a tree-type protocol devised by Capetanakis (1979). In this algorithm, users are not allowed to transmit new packets that are generated until all earlier collisions are resolved. A user can transmit a new packet in a time slot immediately following its generation, provided that all previous packets that have collided have been transmitted successfully. If a new packet is generated while the channel is clearing the previous collisions, the packet is stored in a buffer. When a new packet collides with another, each user assigns its respective packet to one of two sets, say A or B, with equal probability (by flipping a coin). Then, if a packet is put in set A, the user transmits it in the next time slot. If it collides again, the user will again randomly assign the packet to one of two sets and the process of transmission is repeated. This process continues until all packets contained in set A are transmitted successfully. Then, all packets in set B are transmitted following the same procedure. All the users monitor the state of the channel, and, hence, they know when all the collisions have been serviced.

When the channel becomes available for transmission of new packets, the earliest generated packets are transmitted first. To establish a queue, the time scale is subdivided into subintervals of sufficiently short duration such that, on average, approximately one packet is generated by a user in a subinterval. Thus, each packet has a "time tag" that is associated with the subinterval in which it was generated. Then, a new packet belonging to the first subinterval is transmitted in the first available time slot. If there is no collision, then a packet from the second subinterval is transmitted, and so on. This procedure continues as new packets are generated and as long as any backlog of packets for transmission exists. Capetanakis has demonstrated that this channel access protocol achieves a maximum throughput of 0.43 packets per slot.

In addition to throughput, another important performance measure in a random access system is the average transmission delay in transmitting a packet. In an ALOHA system, the average number of transmissions per packet is G/S. To this number we may add the average waiting time between transmissions and, thus, obtain an average delay for a successful transmission. We recall from the above discussion that in the Abramson protocol, the parameter α determines the average delay between retransmissions. If we select α small, we obtain the desirable effect of smoothing out the channel load at times of peak loading, but the result is a long retransmission delay. This is the trade-off in the selection of α in Equation 16.5–2. On the other hand, the Capetanakis protocol has been shown to have a smaller average delay in the transmission of packets. Hence, it outperforms Abramson's protocol in both average delay and throughput.

Another important issue in the design of random access protocols is the stability of the protocol. In our treatment of ALOHA-type channel access protocols, we implicitly assumed that for a given offered load, an equilibrium point is reached where the average number of packets entering the channel is equal to the average number of packets transmitted successfully. In fact, it can be demonstrated that any channel access protocol, such as the Abramson protocol, that does not take into account the number of previous unsuccessful transmissions in establishing a retransmission policy is inherently unstable. On the other hand, the Capetanakis algorithm differs from the Abramson protocol in this respect and has been proved to be stable. A thorough discussion of the stability issues of random access protocols is found in the paper by Massey (1988).

16.5–2 Carrier Sense Systems and Protocols

As we have observed, ALOHA-type (slotted and unslotted) random access protocols yield relatively low throughput. Furthermore, a slotted ALOHA system requires that users transmit at synchronized time slots. In channels where transmission delays are relatively small, it is possible to design random access protocols that yield higher throughput. An example of such a protocol is *carrier sensing* with collision detection, which is used as a standard Ethernet protocol in local area networks. This protocol is generally known as *carrier sense multiple access with collision detection* (CSMA/CD).

The CSMA/CD protocol is simple. All users listen for transmissions on the channel. A user who wishes to transmit a packet seizes the channel when it senses that the channel is idle. Collisions may occur when two or more users sense an idle channel and begin transmission. When the users that are transmitting simultaneously sense a collision, they transmit a special signal, called a *jam signal*, that serves to notify all users of the collision and abort their transmissions. Both the carrier sensing feature and the abortion of transmission when a collision occurs result in minimizing the channel downtime and, hence, yield a higher throughput.

To elaborate on the efficiency of CSMA/CD, let us consider a local area network having a bus architecture, as shown in Figure 16.5–4. Consider two users U_1 and U_2 at the maximum separation, i.e., at the two ends of the bus, and let τ_d be the propagation

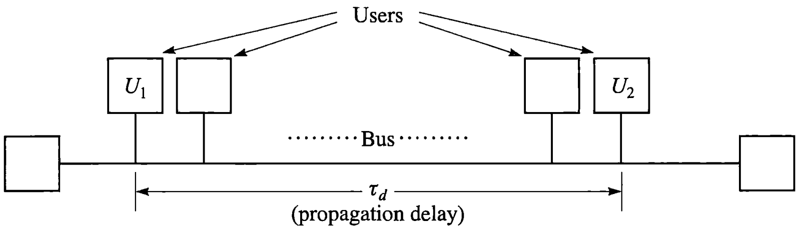


FIGURE 16.5–4
Local area network with bus architecture.

delay for a signal to travel the length of the bus. Then, the (maximum) time required to sense an idle channel is τ_d. Suppose that U_1 transmits a packet of duration T_p. User U_2 may seize the channel τ_d seconds later by using carrier sensing and begins to transmit. However, user U_1 would not know of this transmission until τ_d seconds after U_2 begins transmission. Hence, we may define the time interval $2\tau_d$ as the (maximum) time interval to detect a collision. If we assume that the time required to transmit the jam signal is negligible, the CSMA/CD protocol yields a high throughput when $2\tau_d \ll T_p$.

There are several possible protocols that may be used to reschedule transmissions when a collision occurs. One protocol is called *nonpersistent CSMA*, a second is called *1-persistent CSMA*, and a generalization of the latter is called p-persistent CSMA.

Nonpersistent CSMA In this protocol, a user that has a packet to transmit senses the channel and operates according to the following rule.

(a) If the channel is idle, the user transmits a packet.
(b) If the channel is sensed busy, the user schedules the packet transmission at a later time according to some delay distribution. At the end of the delay interval, the user again senses the channel and repeats steps (a) and (b).

1-Persistent CSMA This protocol is designed to achieve high throughput by not allowing the channel to go idle if some user has a packet to transmit. Hence, the user senses the channel and operates according to the following rule.

(a) If the channel is sensed idle, the user transmits the packet with probability 1.
(b) If the channel is sensed busy, the user waits until the channel becomes idle and transmits a packet with probability one. Note that in this protocol, a collision will always occur when more than one user has a packet to transmit.

p-Persistent CSMA To reduce the rate of collisions in 1-persistent CSMA and increase the throughput, we should randomize the starting time for transmission of packets. In particular, upon sensing that the channel is idle, a user with a packet to transmit sends it with probability p and delays it by τ with probability $1 - p$. The probability p is chosen in a way that reduces the probability of collisions while the idle periods between consecutive (non-overlapping) transmissions is kept small. This is accomplished by subdividing the time axis into minislots of duration τ and selecting the packet transmission at the beginning of a minislot. In summary, in the p-persistent protocol, a user with a packet to transmit proceeds as follows.

(a) If the channel is sensed idle, the packet is transmitted with probability p, and with probability $1 - p$ the transmission is delayed by τ seconds.
(b) If at $t = \tau$, the channel is still sensed to be idle, step (a) is repeated. If a collision occurs, the users schedule retransmission of the packets according to some preselected transmission delay distribution.
(c) If at $t = \tau$, the channel is sensed busy, the user waits until it becomes idle, and the operates as in steps (a) and (b) above.

Slotted versions of the above protocol can also be constructed.

The throughput analysis for the nonpersistent and the p-persistent CSMA/CD protocols has been performed by Kleinroch and Tobagi (1975), based on the following assumptions:

1. The average retransmission delay is large compared with the packet duration T_p.
2. The interarrival times of the point process defined by the start times of all the packets plus retransmissions are independent and exponentially distributed.

For the nonpersistent CSMA, the throughput is

$$S = \frac{Ge^{-aG}}{G(1+2a)+e^{-aG}} \tag{16.5–11}$$

where the parameter $a = \tau_d/T_p$. Note that as $a \to 0$, $S \to G/(1+G)$. Figure 16.5–5 illustrates the throughput versus the offered traffic G, with a as a parameter. We observe that $S \to 1$ as $G \to \infty$ for $a = 0$. For $a > 0$, the value of $S_{\max}$ decreases.

For the 1-persistent protocol, the throughput obtained by Kleinrock and Tobagi (1975) is

$$S = \frac{G[1+G+aG(1+G+\frac{1}{2}aG)]e^{-G(1+2a)}}{G(1+2a)-(1-e^{-aG})+(1+aG)e^{-G(1+a)}} \tag{16.5–12}$$

In this case,

$$\lim_{a\to 0} S = \frac{G(1+G)e^{-G}}{G+e^{-G}} \tag{16.5–13}$$

which has a smaller peak value than the nonpersistent protocol.

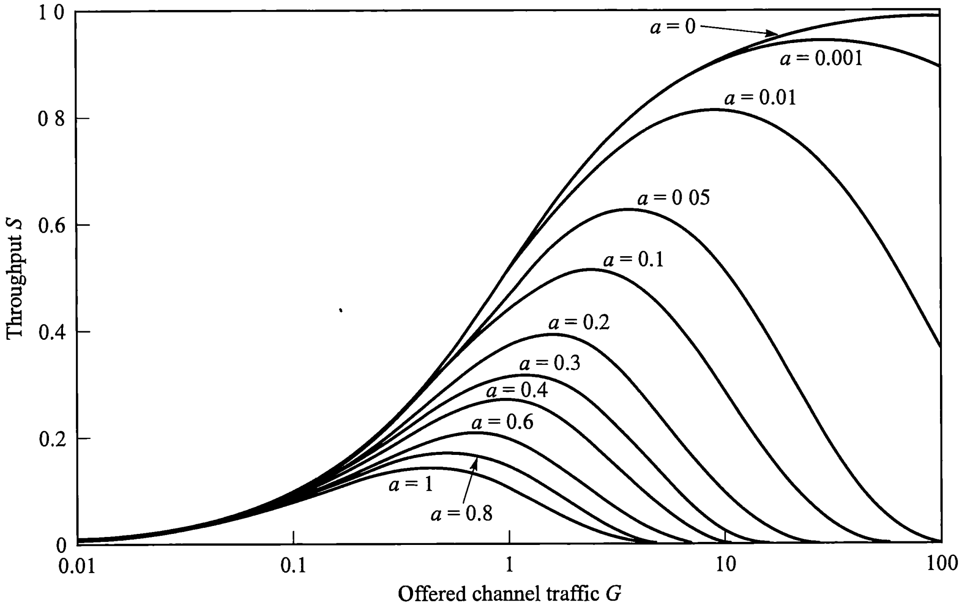


FIGURE 16.5–5
Throughput in nonpersistent CSMA. [*From Kleinrock and Tobagi (1975), © IEEE.*]

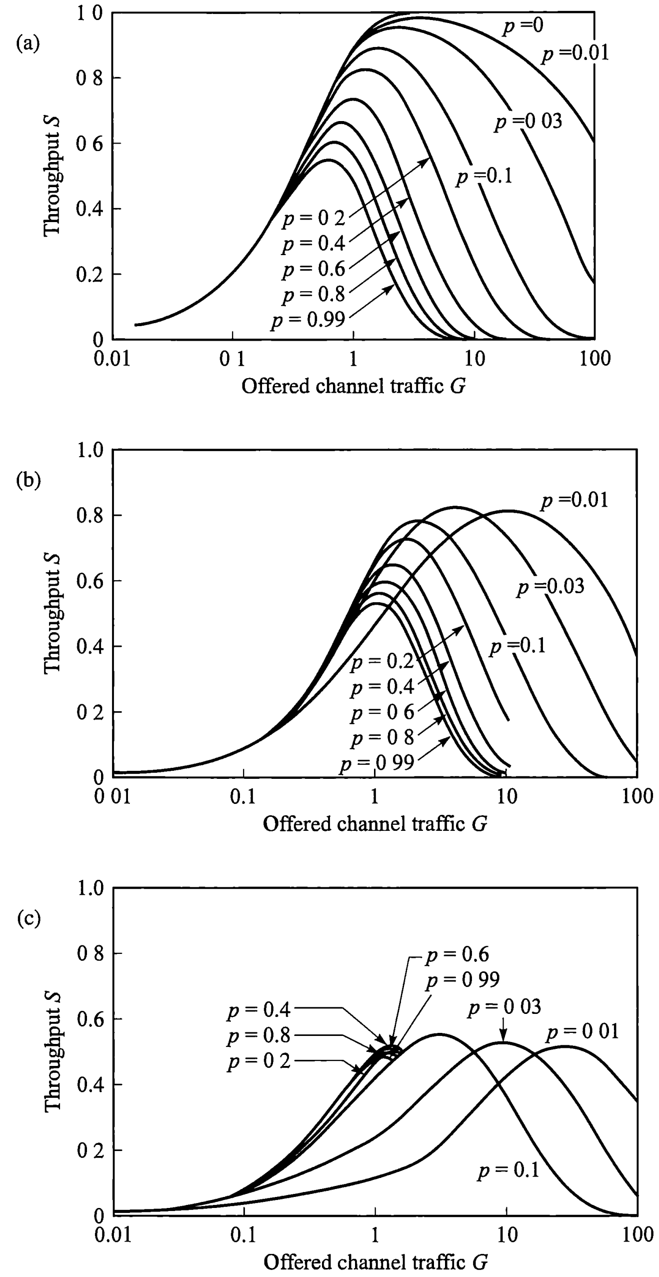


FIGURE 16.5–6

Channel throughput in p-persistent CSMA: (a) $a = 0$; (b) $a = 0.01$; (c) $a = 0.1$. [*From Kleinrock and Tobagi (1975),* © *IEEE.*]

By adopting the p-persistent protocol, it is possible to increase the throughput relative to the 1-persistent scheme. For example, Figure 16.5–6 illustrates the throughput versus the offered traffic with $a = \tau_d/T_p$ fixed and with p as a parameter. We observe that as p increases toward unity, the maximum throughput decreases.

The transmission delay was also evaluated by Kleinrock and Tobagi (1975). Figure 16.5–7 illustrates the graphs of the delay (normalized by T_p) versus the throughput S for the slotted nonpersistent and p-persistent CSMA protocols. Also shown for comparison is the delay versus throughput characteristic of the ALOHA slotted and unslotted protocols. In this simulation, only the newly generated packets are derived independently from a Poisson distribution. Collisions and uniformly distributed random

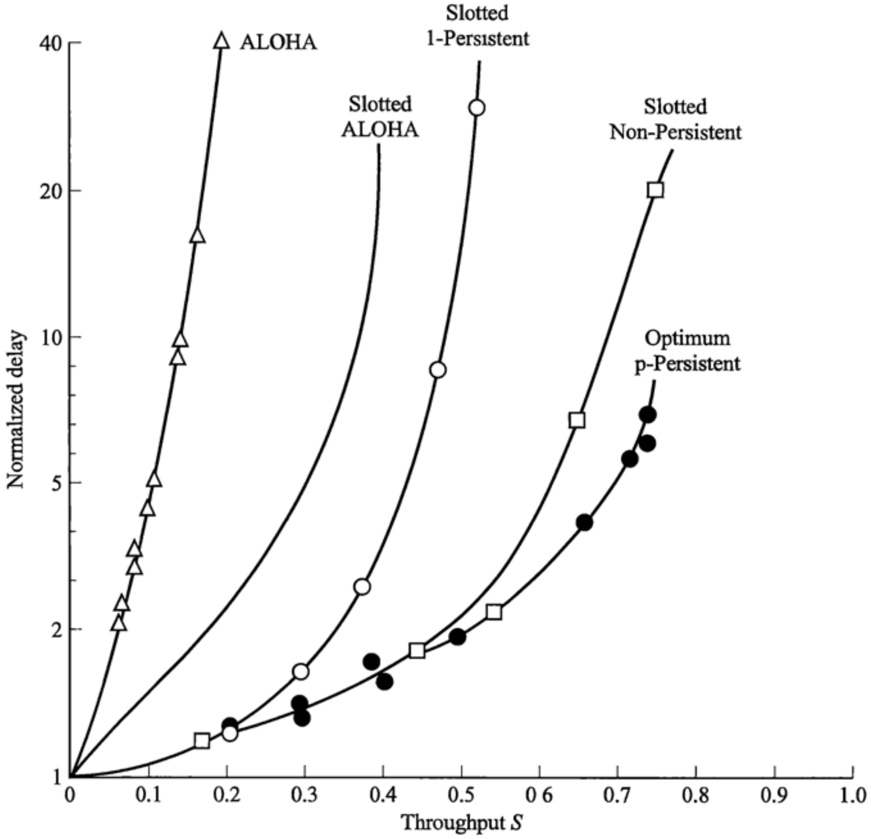


FIGURE 16.5–7
Throughput versus delay from simulation ($a = 0.01$). [*From Kleinrock and Tobagi (1975),* © *IEEE.*]

retransmissions are handled without further assumptions. These simulation results illustrate the superior performance of the p-persistent and the nonpersistent protocols relative to the ALOHA protocols. Note that the graph label "optimum p-persistent" is obtained by finding the optimum value of p for each value of the throughput. We observe that for small values of the throughput, the 1-persistent ($p = 1$) protocol is optimal.

16.6
BIBLIOGRAPHICAL NOTES AND REFERENCES

FDMA was the dominant multiple access scheme that has been used for decades in telephone communication systems for analog voice transmission. With the advent of digital speech transmission using PCM, DPCM, and other speech coding methods, TDMA has replaced FDMA as the dominant multiple access scheme in telecommunications. CDMA and random access methods, in general, have been developed over the past three decades, primarily for use in wireless signal transmission and in local area wireline networks.

Multiuser information theory deals with basic information-theoretic limits in source coding for multiple sources, and channel coding and modulation for multiple access channels. A large amount of literature exists on these topics. In the context of our treatment of multiple access methods, the reader will find the papers by Cover (1972), El Gamal and Cover (1980), Bergmans and Cover (1974), Hui (1984), Cover (1998), and the book by Cover and Thomas (2006) particularly relevant. The capacity of a cellular CDMA system has been considered in the paper by Gilhousen et al. (1991).

Signal demodulation and detection for multiuser communications has received considerable attention in recent years. The reader is referred to the papers by Verdu (1986a,b,c, 1989), Lupas and Verdu (1990), Xie et al. (1990a,b), Poor and Verdu (1988), Zhang and Brady (1993), Madhow and Honig (1994), Zvonar and Brady (1995), Viterbi (1990), Varanasi (1999), and the books by Verdu (1998), Viterbi (1995), and Garg et al. (1997). Earlier work on signal design and demodulation for multiuser communications is found in the papers by Van Etten (1975, 1976), Horwood and Gagliardi (1975), and Kaye and George (1970).

The achievable throughput (capacity) of point-to-multipoint signal transmission employing multiple antennas in a Gaussian broadcast channel has been evaluated in papers published by Yu and Cioffi (2002), Caire and Shamai (2003), Viswanath and Tse (2003), Vishwanath et al. (2003), and Weingarten et al. (2004), as well as in the book by Tse and Viswanath (2005). Various precoding schemes for the MIMO broadcast channel have been considered in several publications, including the papers by Yu and Cioffi (2001), Fisher et al. (2002), Ginis and Cioffi (2002), Windpassinger et al. (2003, 2004a, 2004b), Peel et al. (2005), Hochwald et al. (2005), and Amihood et al. (2006, 2007). The book by Fischer (2002) treats precoding and signal shaping for multichannel digital transmission.

The ALOHA system, which was one of the earliest random access systems, is treated in the papers by Abramson (1970, 1977) and Roberts (1975). These papers contain the throughput analysis for unslotted and slotted systems. More recently, Abramson (1994), considers an ALOHA system that employs spread spectrum signals and provides a link to CDMA systems. Stability issues regarding the ALOHA protocols may be found in the papers by Carleial and Hellman (1975), Ghez et al. (1988), and Massey (1988). Stable protocols based on tree algorithms for random access channels were first given by Capetanakis (1979). The carrier sense multiple access protocols that we described are due to Kleinrock and Tobagi (1975). Finally, we mention the IEEE Press book edited by Abramson (1993), which contains a collection of papers dealing with multiple access communications.

PROBLEMS

16.1 In the formulation of the CDMA signal and channel models described in Section 16.3–1, we assumed that the received signals are real. For $K > 1$, this assumption implies phase synchronism at all transmitters, which is not very realistic in a practical system. To accommodate the case where the carrier phases are not synchronous, we may simply alter the signature waveforms for the K users, given by Equation 16.3–1, to be complex-valued, of the form

$$g_k(t) = e^{j\theta_k} \sum_{n=0}^{L-1} a_k(n) p(t - nT_c), \qquad 1 \le k \le K$$

where θ_k represents the constant phase offset of the kth transmitter as seen by the common receiver.

a. Given this complex-valued form for the signature waveforms, determine the form of the optimum ML receiver that computes the correlation metrics analogous to Equation 16.3–15.

b. Repeat the derivation for the optimum ML detector for asynchronous transmission that is analogous to Equation 16.3–19.

16.2 Consider a TDMA system where each user is limited to a transmitted power P, independent of the number of users. Determine the capacity per user, C_K, and the total capacity KC_K. Plot C_K and KC_K as functions of $\mathcal{E}_b/N_0$ and comment on the results as $K \to \infty$.

16.3 Consider an FDMA system with $K = 2$ users, in an AWGN channel, where user 1 is assigned a bandwidth $W_1 = \alpha W$ and user 2 is assigned a bandwidth $W_2 = (1 - \alpha)W$, where $0 \le \alpha \le 1$. Let P_1 and P_2 be the average powers of the two users.

a. Determine the capacities C_1 and C_2 of the two users and their sum $C = C_1 + C_2$ as a function of α. On a two-dimensional graph of the rates R_2 versus R_1, plot the graph of the points (C_2, C_1) as α varies in the range $0 \le \alpha \le 1$.

b. Recall that the rates of the two users must satisfy the conditions

$$\begin{aligned} R_1 &< W_1 \log_2\left(1 + \frac{P_1}{W_1 N_0}\right) \\ R_2 &< W_2 \log_2\left(1 + \frac{P_2}{W_2 N_0}\right) \\ R_1 + R_2 &< W \log_2\left(1 + \frac{P_1 + P_2}{W N_0}\right) \end{aligned}$$

Determine the total capacity C when $P_1/\alpha = P_2/(1-\alpha) = P_1 + P_2$, and, thus, show that the maximum rate is achieved when $\alpha/(1 - \alpha) = P_1/P_2 = W_1/W_2$.

16.4 Consider a TDMA system with $K = 2$ users in an AWGN channel. Suppose that the two transmitters are peak-power-limited to P_1 and P_2, and let user 1 transmit for 100α percent of the available time and user 2 transmit $100(1 - \alpha)$ percent of the time. The available bandwidth is W.

a. Determine the capacities C_1, C_2, and $C = C_1 + C_2$ as functions of α.

b. Plot the graph of the points (C_2, C_1) as α varies in the range $0 \le \alpha \le 1$.

16.5 Consider a TDMA system with $K = 2$ users in an AWGN channel. Suppose that the two transmitters are average-power-limited, with powers P_1 and P_2. User 1 transmits $100a$ percent of the time and user 2 transmits $100(1 - \alpha)$ percent of the time. The channel bandwidth is W.

a. Determine the capacities C_1, C_2, and $C = C_1 + C_2$ as functions of α.

b. Plot the graph of the points (C_2, C_1) as α varies in the range $0 \le \alpha \le 1$.

c. What is the similarity between this solution and the FDMA system in Problem 16.3?

16.6 Consider a two-user, synchronous CDMA transmission system, where the received signal is

$$r(t) = \sqrt{\mathcal{E}_1}b_1 g_1(t) + \sqrt{\mathcal{E}_2}b_2 g_2(t) + n(t), \qquad 0 \le t \le T$$

and $(b_1, b_2) = (\pm 1, \pm 1)$. The noise process $n(t)$ is zero-mean Gaussian and white, with spectral density $N_0/2$. The demodulator for $r(t)$ is shown in Figure P16.6.

a. Show that the correlator outputs r_1 and r_2 at $t = T$ may be expressed as

$$\begin{aligned} r_1 &= \sqrt{\mathcal{E}_1}b_1 + \sqrt{\mathcal{E}_2}\rho b_2 + n_1 \\ r_2 &= \sqrt{\mathcal{E}_1}b_1\rho + \sqrt{\mathcal{E}_2}b_2 + n_2 \end{aligned}$$

b. Determine the variances of n_1 and n_2 and the covariance of n_1 and n_2.

c. Determine the joint PDF $p(r_1, r_2 | b_1, b_2)$.

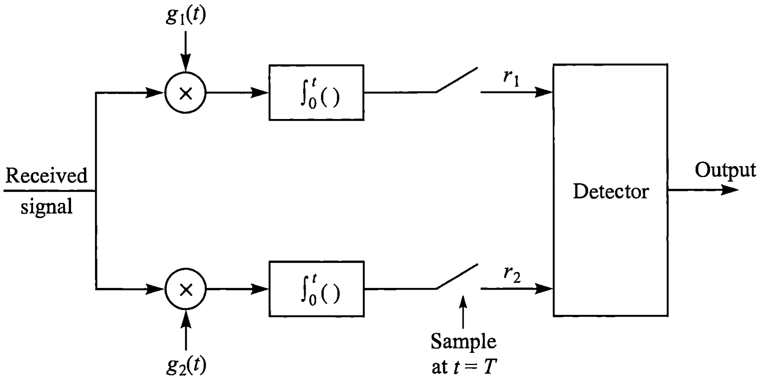


FIGURE P16.6

16.7 Consider the two-user, synchronous CDMA transmission system described in Problem 16.6. The conventional single-user detector for the information bits b_1 and b_2 gives the outputs

$$\begin{aligned} b_1 &= \text{sgn}(r_1) \\ b_2 &= \text{sgn}(r_2) \end{aligned}$$

Assuming that $P(b_1 = 1) = P(b_2 = 1) = \frac{1}{2}$, and b_1 and b_2 are statistically independent, determine the probability of error for this detector.

16.8 Consider the two-user, synchronous CDMA transmission system described in Problem 16.6. $P(b_1 = 1) = P(b_2 = 1) = \frac{1}{2}$ and $P(b_1, b_2) = P(b_1)P(b_2)$. The *jointly optimum detector* makes decisions based on the maximum a posteriori probability (MAP) criterion. That is, the detector computes

$$\max_{b_1, b_2} P[b_1, b_2 | r(t), 0 \le t \le T]$$

a. For the equally likely information bits (b_1, b_2) show that the MAP criterion is equivalent to the maximum-likelihood (ML) criterion

$$\max_{b_1, b_2} p[r(t), 0 \le t \le T | b_1, b_2]$$

b. Show that the ML criterion in (a) leads to the jointly optimum detector that makes decisions on b_1 and b_2 according to the following rule:

$$\max_{b_1,b_2}\left(\sqrt{\mathcal{E}_1}b_1r_1+\sqrt{\mathcal{E}_2}b_2r_2-\sqrt{\mathcal{E}_1\mathcal{E}_2}\rho b_1b_2\right)$$

16.9 Consider the two-user, synchronous CDMA transmission system described in Problem 16.6. $P(b_1=1)=P(b_2=1)=\frac{1}{2}$ and $P(b_1,b_2)=P(b_1)P(b_2)$. The individually optimum detector makes decisions based on the MAP criterion. That is, the detector computes the a posteriori probabilities.

$$\begin{aligned}P[b_1|r(t),0\le t\le T]&=P[b_1,b_2=1|r(t),\ 0\le t\le T]\\&\quad+P[b_1,b_2=-1|r(t),\ 0\le t\le T]\end{aligned}$$

and

$$\begin{aligned}P[b_2|r(t),0\le t\le T]&=P[b_1=1,b_2|r(t),\ 0\le t\le T]\\&\quad+P[b_1=-1,b_2|r(t),\ 0\le t\le T]\end{aligned}$$

a. Show that an equivalent test statistic for this individually optimum MAP detector for the information bit b_1 is

$$\max_{b_1}\left\{\frac{\sqrt{\mathcal{E}_1}r_1}{N_0}b_1+\ln\cosh\left(\frac{\sqrt{\mathcal{E}_2}r_2-\sqrt{\mathcal{E}_1\mathcal{E}_2}\rho b_1}{N_0}\right)\right\}$$

b. By substituting $b_1=1$ and $b_1=-1$ into the expression in (a), show that the test statistic in (a) is equivalent to selecting b_1 according to the relation

$$\hat{b}_1=\operatorname{sgn}\left[r_1-\frac{N_0}{2\sqrt{\mathcal{E}_1}}\ln\frac{\cosh\left(\sqrt{\mathcal{E}_2}r_2+\sqrt{\mathcal{E}_1\mathcal{E}_2}\rho\right)/N_0}{\cosh\left(\sqrt{\mathcal{E}_2}r_2-\sqrt{\mathcal{E}_1\mathcal{E}_2}\rho\right)/N_0}\right]$$

16.10 Show that the asymptotic efficiency of the conventional single-user detector in a CDMA system with K users transmitting synchronously is

$$\eta_k=\left[\max\left\{0,1-\sum_{j\ne k}\sqrt{\frac{\mathcal{E}_j}{\mathcal{E}_k}}|\rho_{jk}(0)|\right\}\right]^2$$

16.11 Consider the jointly optimum detector defined in Problem 16.8 for the two-user, synchronous CDMA system. Show that the (symbol) error probability for this detector may be upper-bounded as

$$Pe<Q\left(\sqrt{\frac{2\mathcal{E}_1}{N_0}}\right)+\frac{1}{2}Q\left(\sqrt{\frac{\mathcal{E}_1+\mathcal{E}_2-2\sqrt{\mathcal{E}_1\mathcal{E}_2}|\rho|}{N_0/2}}\right)$$

16.12 Consider the jointly optimum detector defined in Problem 16.8 for the two-user, synchronous CDMA system.

a. Show that the asymptotic efficiency for this detector for user 1

$$\eta_1=\min\left\{1,1+\frac{\mathcal{E}_2}{\mathcal{E}_1}-2\sqrt{\frac{\mathcal{E}_2}{\mathcal{E}_1}}|\rho|\right\}$$

b. Plot and compare the asymptotic efficiencies of the jointly optimum detector and the conventional single-user detector for $\rho = 0.1$ and $\rho = 0.2$.

16.13 Consider the two-user synchronous CDMA system in Problem 16.6. Determine the probability of error for each user that employs a decorrelating detector when $\mathcal{E}_1 \neq \mathcal{E}_2$.

16.14 Consider a two-user synchronous CDMA system where the received signal is given in Problem 16.6. Each user employs the minimum MSE detector specified by Equations 16.3–51 to 16.3–53.

a. Determine the linear transformation matrix A^0 for the two users.

b. Show that the MMSE detector approaches the decorrelating detector as $N_0 \to 0$.

c. Show that the MMSE detector approaches the conventional single-user detector as $N_0 \to \infty$.

16.15 Consider the asynchronous communication system shown in Figure P16.15. The two receivers are not colocated, and the white noise processes $n^{(1)}(t)$ and $n^{(2)}(t)$ may be considered to be independent. The noise processes are identically distributed, with power spectral density σ^2 and zero-mean. Since the receivers are not colocated, the relative delays between the users are not the same—denote the relative delay of user k at receiver i by $\tau_k^{(i)}$. All other signal parameters coincide for the receivers, and the received signal at receiver i is

$$r^{(i)}(t) = \sum_{k=1}^{2} \sum_{l=-\infty}^{\infty} b_k(l) s_k\left(t - lT - \tau_k^{(i)}\right) + n^{(i)}(t)$$

where s_k has support on $[0, T]$. You may assume that the receiver i has full knowledge of the waveforms, energies, and relative delays $\tau_1^{(i)}$ and $\tau_2^{(i)}$. Although receiver i is eventually interested only in the data from transmitter i, note that there is a free communication link between the sampler of one receiver, and the postprocessing circuitry of the other. Following each postprocessor, the decision is attained by threshold detection. In this problem, you will consider options for postprocessing and for the communication link in order to improve performance.

a. What is the bit error probability for users 1 and 2 of a receiver pair that does not utilize the communication link and does not perform postprocessing? Use the following

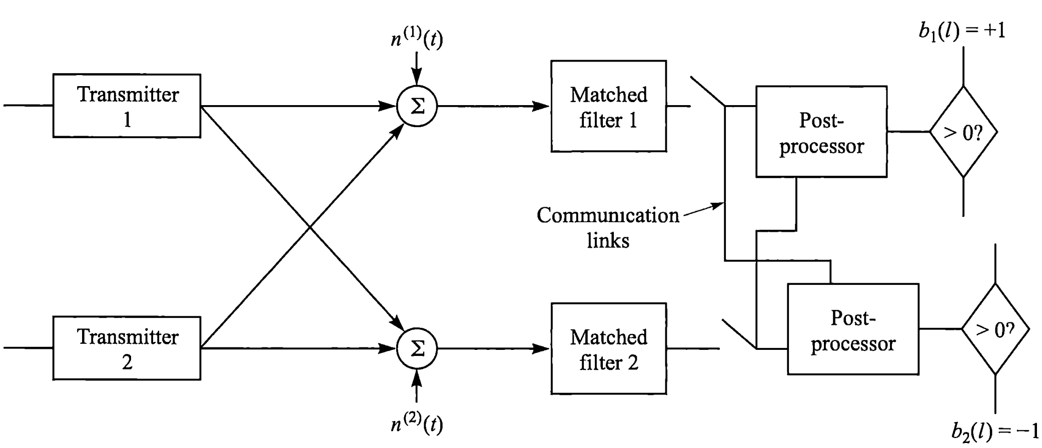


FIGURE P16.15

notation:

$$y_k(l) = \int s_k\left(t - lT - \tau_k^{(k)}\right) r^{(k)}(t)\, dt$$

$$\rho_{12}^{(i)} = \int s_1\left(t - \tau_1^{(i)}\right) s_2\left(t - \tau_2^{(i)}\right) dt$$

$$\rho_{21}^{(i)} = \int s_1\left(t - \tau_1^{(i)}\right) s_2\left(t + T - \tau_2^{(i)}\right) dt$$

$$w_k = \int s_k^2\left(t - \tau_k^{(1)}\right) dt = \int s_k^2\left(t - \tau_k^{(2)}\right) dt$$

b. Consider a postprocessor for receiver 1 that accepts $y_2(l-1)$ and $y_2(l)$ from the communication link and implements the following postprocessing on $y_1(l)$

$$z_l(l) = y_1(l) - \rho_{21}^{(1)}\mathrm{sgn}[y_2(l-1)] - \rho_{12}^{(1)}\mathrm{sgn}[y_2(l)].$$

Determine an exact expression for the bit error rate for user 1.

c. Determine the asymptotic multiuser efficiency of the receiver proposed in (b), and compare with that in (a). Does this receiver always perform better than that proposed in (a)?

16.16 In a pure ALOHA system, the channel bit rate is 2400 bits/s. Suppose that each terminal transmits a 100-bit message every minute on the average.

a. Determine the maximum number of terminals that can use the channel.

b. Repeat (a) if slotted ALOHA is used.

16.17 An alternative derivation for the throughput in a pure ALOHA system may be obtained from the relation $G = S + A$, where A is the average (normalized) rate of retransmissions. Show that $A = G(1 - e^{-2G})$ and then solve for S.

16.18 For a Poisson process, the probability of k arrivals in a time interval T is

$$P(k) = \frac{e^{-\lambda T}(\lambda T)^k}{k!}, \qquad k = 0, 1, 2, \ldots$$

a. Determine the average number of arrivals in the interval T.

b. Determine the variance σ^2 in the number of arrivals in the interval T.

c. What is the probability of at least one arrival in the interval T?

d. What is the probability of exactly one arrival in the interval T?

16.19 Refer to Problem 16.18. The average arrival rate is $\lambda = 10$ packets/s. Determine

a. The average time between arrivals.

b. The probability that another packet will arrive within 1 s; within 100 ms.

16.20 Consider a pure ALOHA system that is operating with a throughput $S = 0.1$ and packets are generated with a Poisson arrival rate λ. Determine

a. The value of G.

b. The average number of attempted transmissions to send a packet.

16.21 Consider a CSMA/CD system in which the transmission rate on the bus is 10 Mbits/s. The bus is 2 km and the propagation delay is 5 μs/km. Packets are 1000 bits long. Determine

a. The end-to-end delay τ_d.
b. The packet duration T_p.
c. The ratio τ_d / T_p.
d. The maximum utilization of the bus and the maximum bit rate.

16.22 Consider an MA communication system with $K = 2$ users and an AWGN channel. The receiver decodes the two signals by preforming SIC. The signal power levels for the two users at the receiver are P_1 and P_2.

a. Suppose that the receiver decodes the signal for user 2 and subtracts signal 2 from the received signal. Then the receiver decodes the signal from user 1 without interference. Determine the maximum rates that can be achieved by users 1 and 2.
b. Now suppose that $P_1 = 10P_2$ and that the signal from user 2 is decoded first. Determine the sum capacity of the two-user system.
c. Repeat part 2 if user 1 is decoded first, and compare the sum capacities in parts b and c.

APPENDIX A

Matrices

A matrix is a rectangular array of real or complex numbers called the elements of the matrix. An $n \times m$ matrix has n rows and m columns. If $m = n$, the matrix is called a square matrix. An n-dimensional vector may be viewed as an $n \times 1$ matrix. An $n \times m$ matrix may be viewed as having n m-dimensional vectors as its rows or m n-dimensional vectors as its columns.

The complex conjugate and the transpose of a matrix $\boldsymbol{A}$ are denoted as $\boldsymbol{A}^*$ and $\boldsymbol{A}^t$, respectively. The conjugate transpose of a matrix with complex elements is denoted as $\boldsymbol{A}^H$; that is, $\boldsymbol{A}^H = [\boldsymbol{A}^*]^t = [\boldsymbol{A}^t]^*$.

A square matrix $\boldsymbol{A}$ is said to be *symmetric* if $\boldsymbol{A}^t = \boldsymbol{A}$. A square matrix $\boldsymbol{A}$ with complex elements is said to be *Hermitian* if $\boldsymbol{A}^H = \boldsymbol{A}$. If $\boldsymbol{A}$ is a square matrix, then $\boldsymbol{A}^{-1}$ designates the inverse of $\boldsymbol{A}$ (if one exists), having the property that

$$\boldsymbol{A}^{-1}\boldsymbol{A} = \boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}_n \qquad \text{(A–1)}$$

where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix, i.e., a square matrix whose diagonal elements are unity and off-diagonal elements are zero. If $\boldsymbol{A}$ has no inverse, it is said to be *singular*.

The *trace* of a square matrix $\boldsymbol{A}$ is denoted as $\text{tr}(\boldsymbol{A})$ and is defined as the sum of the diagonal elements, i.e.,

$$\text{tr}(\boldsymbol{A}) = \sum_{i=1}^{n} a_{ii} \qquad \text{(A–2)}$$

The rank of an $n \times m$ matrix $\boldsymbol{A}$ is the maximum number of linearly independent columns or rows in the matrix (it makes no difference whether we take rows or columns). A matrix is said to be of *full rank* if its rank is equal to the number of rows or columns, whichever is smaller.

The following are some additional matrix properties (lowercase letters denote vectors):

$$\begin{aligned} (\boldsymbol{A}\boldsymbol{v})^t &= \boldsymbol{v}^t\boldsymbol{A}^t \qquad & (\boldsymbol{A}\boldsymbol{B})^{-1} &= \boldsymbol{B}^{-1}\boldsymbol{A}^{-1} \\ (\boldsymbol{A}\boldsymbol{B})^t &= \boldsymbol{B}^t\boldsymbol{A}^t \qquad & (\boldsymbol{A}^t)^{-1} &= (\boldsymbol{A}^{-1})^t \end{aligned} \qquad \text{(A–3)}$$

A.1

EIGENVALUES AND EIGENVECTORS OF A MATRIX

Let $\boldsymbol{A}$ be an $n \times n$ square matrix. A nonzero vector $\boldsymbol{v}$ is called an eigenvector of $\boldsymbol{A}$ and λ is the associated eigenvalue if

$$\boldsymbol{A}\boldsymbol{v} = \lambda\boldsymbol{v} \tag{A–4}$$

If $\boldsymbol{A}$ is a Hermitian $n \times n$ matrix, then there exist n mutually orthogonal eigenvectors $\boldsymbol{v}_i$, $i = 1, 2, \ldots, n$. Usually, we normalize each eigenvector to unit length, so that

$$\boldsymbol{v}_i^H \boldsymbol{v}_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{A–5}$$

In such a case, the eigenvectors are orthonormal.

We define an $n \times n$ matrix $\boldsymbol{Q}$ whose ith column is the eigenvector $\boldsymbol{v}_i$. Then

$$\boldsymbol{Q}^H \boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{Q}^H = \boldsymbol{I}_n \tag{A–6}$$

Furthermore, $\boldsymbol{A}$ may be represented (decomposed) as

$$\boldsymbol{A} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^H \tag{A–7}$$

where $\boldsymbol{\Lambda}$ is an $n \times n$ diagonal matrix with elements equal to the eigenvalues of $\boldsymbol{A}$. This decomposition is called a *spectral decomposition* of a Hermitian matrix.

If $\boldsymbol{u}$ is an $n \times 1$ nonzero vector for which $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{0}$, then $\boldsymbol{u}$ is called a null vector of $\boldsymbol{A}$. When $\boldsymbol{A}$ is Hermitian and $\boldsymbol{A}\boldsymbol{u} = \boldsymbol{0}$ for some vector $\boldsymbol{u}$, then $\boldsymbol{A}$ is singular. A singular Hermitian matrix has at least one zero eigenvalue.

Now, consider the scalar quadratic form $\boldsymbol{u}^H\boldsymbol{A}\boldsymbol{u}$ associated with the Hermitian matrix $\boldsymbol{A}$. If $\boldsymbol{u}^H\boldsymbol{A}\boldsymbol{u} > 0$, the matrix $\boldsymbol{A}$ is said to be positive definite. In such a case, all the eigenvalues of $\boldsymbol{A}$ are positive. On the other hand, if $\boldsymbol{u}^H\boldsymbol{A}\boldsymbol{u} \geq 0$, matrix $\boldsymbol{A}$ is said to be positive semidefinite. In such a case, all the eigenvalues of $\boldsymbol{A}$ are nonnegative.

The following properties involving the eigenvalues of an arbitrary $n \times n$ matrix $\boldsymbol{A} = (a_{ij})_n$ hold:

$$\sum_{i=1}^{n} \lambda_i = \sum_{i=1}^{n} a_{ii} = \operatorname{tr}(\boldsymbol{A}) \tag{A–8}$$

$$\prod_{i=1}^{n} \lambda_i = \det(\boldsymbol{A}) \tag{A–9}$$

$$\sum_{i=1}^{n} \lambda_i^k = \operatorname{tr}(\boldsymbol{A}^k) \tag{A–10}$$

$$\operatorname{tr}(\boldsymbol{A}^t\boldsymbol{A}) = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2 \geq \sum_{i=1}^{n} \lambda_i^2, \qquad \boldsymbol{A} \text{ real} \tag{A–11}$$

A.2
SINGULAR-VALUE DECOMPOSITION

The singular-value decomposition (SVD) is another orthogonal decomposition of a matrix. Let us assume that $\boldsymbol{A}$ is an $n \times m$ matrix of rank r. Then there exist an $n \times r$ matrix $\boldsymbol{U}$, an $m \times r$ matrix $\boldsymbol{V}$, and an $r \times r$ diagonal matrix Σ such that $\boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{V}^H\boldsymbol{V} = \boldsymbol{I}_r$ and

$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^H \tag{A–12}$$

where $\boldsymbol{\Sigma} = \text{diag}\,(\sigma_1, \sigma_2, \ldots, \sigma_r)$. The r diagonal elements of $\boldsymbol{\Sigma}$ are strictly positive and are called the *singular values* of matrix $\boldsymbol{A}$. For convenience, we assume that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r$.

The SVD of matrix $\boldsymbol{A}$ may be expressed as

$$\boldsymbol{A} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^H \tag{A–13}$$

where $\boldsymbol{u}_i$ are the column vectors of $\boldsymbol{U}$, which are called the *left singular vectors* of $\boldsymbol{A}$, and $\boldsymbol{v}_i$ are the column vectors of $\boldsymbol{V}$, which are called the right singular vectors of $\boldsymbol{A}$.

The singular values $\{\sigma_i\}$ are the nonnegative square roots of the eigenvalues of matrix $\boldsymbol{A}^H\boldsymbol{A}$. To demonstrate this, we postmultiply Equation A–12 by $\boldsymbol{V}$. Thus, we obtain

$$\boldsymbol{AV} = \boldsymbol{U\Sigma} \tag{A–14}$$

or, equivalently,

$$\boldsymbol{Av}_i = \sigma_i \boldsymbol{u}_i, \qquad i = 1, 2, \ldots, r \tag{A–15}$$

Similarly, we postmultiply $\boldsymbol{A}^H = \boldsymbol{V\Sigma U}^H$ by $\boldsymbol{U}$. Thus, we obtain

$$\boldsymbol{A}^H\boldsymbol{U} = \boldsymbol{V\Sigma} \tag{A–16}$$

or, equivalently,

$$\boldsymbol{A}^H\boldsymbol{u}_i = \sigma \boldsymbol{v}_i, \qquad i = 1, 2, \ldots, r \tag{A–17}$$

Then, by premultiplying both sides of Equation A–15 with $\boldsymbol{A}^H$ and using Equation A–17, we obtain

$$\boldsymbol{A}^H\boldsymbol{Av}_i = \sigma_i^2 \boldsymbol{v}_i, \qquad i = 1, 2, \ldots, r \tag{A–18}$$

This demonstrates that the r nonzero eigenvalues of $\boldsymbol{A}^H\boldsymbol{A}$ are the squares of the singular values of $\boldsymbol{A}$, and the corresponding r eigenvectors $\boldsymbol{v}_i$ are the right singular vectors of $\boldsymbol{A}$. The remaining $m-r$ eigenvalues of $\boldsymbol{A}^H\boldsymbol{A}$ are zero. On the other hand, if we premultiply both sides of Equation A–17 by $\boldsymbol{A}$ and use Equation A–15, we obtain

$$\boldsymbol{AA}^H\boldsymbol{u}_i = \sigma_i^2 \boldsymbol{u}_i, \qquad i = 1, 2, \ldots, r \tag{A–19}$$

This demonstrates that the r nonzero eigenvalues of $\boldsymbol{AA}^H$ are the squares of the singular values of $\boldsymbol{A}$, and the corresponding r eigenvectors $\boldsymbol{u}_i$ are the left singular vectors of $\boldsymbol{A}$. The remaining $n-r$ eigenvalues of $\boldsymbol{AA}^H$ are zero. Hence, $\boldsymbol{AA}^H$ and $\boldsymbol{A}^H\boldsymbol{A}$ have the same set of nonzero eigenvalues.

A.3

MATRIX NORM AND CONDITION NUMBER

Recall that the *Euclidean norm* (L_2 norm) of a vector $\boldsymbol{v}$, denoted as $\|\boldsymbol{v}\|$, is defined as

$$\|\boldsymbol{v}\| = (\boldsymbol{v}^H \boldsymbol{v})^{1/2} \tag{A–20}$$

The Euclidean norm of a matrix $\boldsymbol{A}$, denoted as $\|\boldsymbol{A}\|$, is defined as

$$\|\boldsymbol{A}\| = \max \frac{\|\boldsymbol{A}\boldsymbol{v}\|}{\|\boldsymbol{v}\|} \tag{A–21}$$

for any vector $\boldsymbol{v}$. It is easy to verify that the norm of a Hermitian matrix is equal to the largest eigenvalue.

Another useful quantity associated with a matrix $\boldsymbol{A}$ is the nonzero minimum value of $\|\boldsymbol{A}\boldsymbol{v}\|/\|\boldsymbol{v}\|$. When $\boldsymbol{A}$ is a nonsingular Hermitian matrix, this minimum value is equal to the smallest eigenvalue.

The squared Frobenius norm of an $n \times m$ matrix $\boldsymbol{A}$ is defined as

$$\|\boldsymbol{A}\|_F^2 = tr(\boldsymbol{A}\boldsymbol{A}^H) = \sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}|^2 \tag{A–22}$$

From the SVD of the matrix $\boldsymbol{A}$, it follows that

$$\|\boldsymbol{A}\|_F^2 = \sum_{i=1}^{n} \lambda_i \tag{A–23}$$

where $\{\lambda_i\}$ are the eigenvalues of $\boldsymbol{A}\boldsymbol{A}^H$.

The following are bounds on matrix norms:

$$\begin{aligned} &\|\boldsymbol{A}\| > 0,\ \boldsymbol{A} \neq \boldsymbol{0} \\ &\|\boldsymbol{A} + \boldsymbol{B}\| \leq \|\boldsymbol{A}\| + \|\boldsymbol{B}\| \\ &\|\boldsymbol{A}\boldsymbol{B}\| \leq \|\boldsymbol{A}\|\,\|\boldsymbol{B}\| \end{aligned} \tag{A–24}$$

The *condition number* of a matrix $\boldsymbol{A}$ is defined as the ratio of the maximum value to the minimum value of $\|\boldsymbol{A}\boldsymbol{v}\|/\|\boldsymbol{v}\|$. When $\boldsymbol{A}$ is Hermitian, the condition number is $\lambda_{\max}/\lambda_{\min}$, where $\lambda_{\max}$ is the largest eigenvalue and $\lambda_{\min}$ is the smallest eigenvalue of $\boldsymbol{A}$.

A.4

THE MOORE–PENROSE PSEUDOINVERSE

Let us consider a rectangular $n \times m$ matrix $\boldsymbol{A}$ of rank r, having an SVD as $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H$. The Moore–Penrose pseudoinverse, denoted by $\boldsymbol{A}^+$, is an $m \times n$ matrix defined as

$$\boldsymbol{A}^+ = \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^H \tag{A–25}$$

where $\boldsymbol{\Sigma}^{-1}$ is an $r \times r$ diagonal matrix with diagonal elements $1/\sigma_i$, $i = 1, 2, \ldots, r$. We may also express $\boldsymbol{A}^+$ as

$$\boldsymbol{A}^+ = \sum_{i=1}^{r} \frac{1}{\sigma_i} \boldsymbol{v}_i \boldsymbol{u}_i^H \tag{A–26}$$

We observe that the rank of $\boldsymbol{A}^+$ is equal to the rank of $\boldsymbol{A}$.

When the rank $r = m$ or $r = n$, the pseudoinverse $\boldsymbol{A}^+$ can be expressed as

$$\begin{aligned} \boldsymbol{A}^+ &= \boldsymbol{A}^H(\boldsymbol{A}\boldsymbol{A}^H)^{-1} && r = n \\ \boldsymbol{A}^+ &= (\boldsymbol{A}^H\boldsymbol{A})^{-1}\boldsymbol{A}^H && r = m \\ \boldsymbol{A}^+ &= \boldsymbol{A}^{-1} && r = m = n \end{aligned} \tag{A–27}$$

These relations are equivalent to $\boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{I}_n$ and $\boldsymbol{A}^+\boldsymbol{A} = \boldsymbol{I}_m$.

APPENDIX B

Error Probability for Multichannel Binary Signals

In multichannel communication systems that employ binary signaling for transmitting information over the AWGN channel, the decision variable at the detector can be expressed as a special case of the general quadratic form

$$D = \sum_{k=1}^{L} \left(A|X_k|^2 + B|Y_k|^2 + CX_kY_k^* + C^*X_k^*Y_k \right) \tag{B–1}$$

in complex-valued Gaussian random variables. A, B, and C are constants; X_k and Y_k are a pair of correlated complex-valued Gaussian random variables. For the channels considered, the L pairs $\{X_k, Y_k\}$ are mutually statistically independent and identically distributed.

The probability of error is the probability that $D < 0$. This probability is evaluated below.

The computation begins with the characteristic function, denoted by $\psi_D(jv)$, of the general quadratic form. The probability that $D < 0$, denoted here as the probability of error P_b, is

$$P_b = P(D < 0) = \int_{-\infty}^{0} p(D)\, dD \tag{B–2}$$

where $p(D)$, the probability density function of D, is related to $\psi_D(jv)$ by the Fourier transform, i.e.,

$$p(D) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi_D(jv) e^{-jvD} dv$$

Hence,

$$P_b = \int_{-\infty}^{0} dD \frac{1}{2\pi} \int_{-\infty}^{\infty} \psi_D(jv) e^{-jvD} dv \tag{B–3}$$

Let us interchange the order of integration and carry out first the integration with respect to D. The result is

$$P_b = -\frac{1}{2\pi j}\int_{-\infty+j\varepsilon}^{\infty+j\varepsilon} \frac{\psi_D(jv)}{v}\,dv \tag{B–4}$$

where a small positive number ε has been inserted in order to move the path of integration away from the singularity at $v = 0$ and which must be positive in order to allow for the interchange in the order of integration.

Since D is the sum of statistically independent random variables, the characteristic function of D factors into a product of L characteristic functions, with each function corresponding to the individual random variables d_k, where

$$d_k = A|X_k|^2 + B|Y_k|^2 + CX_kY_k^* + C^*X_k^*Y_k$$

The characteristic function of d_k is

$$\psi_{d_k}(jv) = \frac{v_1v_2}{(v+jv_1)(v-jv_2)}\exp\left[\frac{v_1v_2\left(-v^2\alpha_{1k} + jv\alpha_{2k}\right)}{(v+jv_1)(v-jv_2)}\right] \tag{B–5}$$

where the parameters v_1, v_2, α_{1k}, and α_{2k} depend on the means $\bar{X}_k$ and $\bar{Y}_k$ and the second (central) moments μ_{xx}, μ_{yy}, and μ_{xy} of the complex-valued Gaussian variables X_k and Y_k through the following definitions ($|C|^2 - AB > 0$):

$$\begin{aligned}
v_1 &= \sqrt{w^2 + \frac{1}{4\left(\mu_{xx}\mu_{yy} - |\mu_{xy}|^2\right)\left(|C|^2 - AB\right)}} - w \\
v_2 &= \sqrt{w^2 + \frac{1}{4\left(\mu_{xx}\mu_{yy} - |\mu_{xy}|^2\right)\left(|C|^2 - AB\right)}} + w \\
w &= \frac{A\mu_{xx} + B\mu_{yy} + C\mu_{xy}^* + C^*\mu_{xy}}{4\left(\mu_{xx}\mu_{yy} - |\mu_{xy}|^2\right)\left(|C|^2 - AB\right)} \\
\alpha_{1k} &= 2(|C|^2 - AB)\left(|\bar{X}_k|^2\mu_{yy} + |\bar{Y}_k|^2\mu_{xx} - \bar{X}_k^*\bar{Y}_k\mu_{xy} - \bar{X}_k\bar{Y}_k^*\mu_{xy}^*\right) \\
\alpha_{2k} &= A|\bar{X}_k|^2 + B|\bar{Y}_k|^2 + C\bar{X}_k^*\bar{Y}_k + C^*\bar{X}_k\bar{Y}_k^* \\
\mu_{xy} &= \tfrac{1}{2}E[(X_k - \bar{X}_k)(Y_k - \bar{Y}_k)^*]
\end{aligned} \tag{B–6}$$

Now, as a result of the independence of the random variables d_k, the characteristic function of D is

$$\begin{aligned}
\psi_D(jv) &= \prod_{k=1}^{L}\psi_{d_k}(jv) \\
\psi_D(jv) &= \frac{(v_1v_2)^L}{(v+jv_1)^L(v-jv_2)^L}\exp\left[\frac{v_1v_2\left(jv\alpha_2 - v^2\alpha_1\right)}{(v+jv_1)(v-jv_1)}\right]
\end{aligned} \tag{B–7}$$

where

$$\alpha_1 = \sum_{k=1}^{L}\alpha_{1k}, \qquad \alpha_2 = \sum_{k=1}^{L}\alpha_{2k} \tag{B–8}$$

The result B–7 is substituted for $\psi_D(jv)$ in Equation B–4, and we obtain

$$P_b = -\frac{(v_1 v_2)^L}{2\pi j}\int_{-\infty+j\varepsilon}^{\infty+j\varepsilon} \frac{dv}{v(v+jv_1)^L(v-jv_2)^L}\exp\left[\frac{v_1 v_2(jv\alpha_2 - v^2\alpha_1)}{(v+jv_1)(v-jv_2)}\right] \tag{B–9}$$

This integral is evaluated as follows.

The first step is to express the exponential function in the form

$$\exp\left(-A_1 + \frac{jA_2}{v+jv_1} - \frac{jA_3}{v-jv_2}\right)$$

where one can easily verify that the constants A_1, A_2, and A_3 are given as

$$\begin{aligned} A_1 &= \alpha_1 v_1 v_2 \\ A_2 &= \frac{v_1^2 v_2}{v_1+v_2}(\alpha_1 v_1 + \alpha_2) \\ A_3 &= \frac{v_1 v_2^2}{v_1+v_2}(\alpha_1 v_2 - \alpha_2) \end{aligned} \tag{B–10}$$

Second, a conformal transformation is made from the v plane onto the p plane via the change in variable

$$p = -\frac{v_1}{v_2}\frac{v-jv_2}{v+jv_1} \tag{B–11}$$

In the p plane, the integral given by Equation B–9 becomes

$$P_b = \frac{\exp\left[v_1 v_2(-2\alpha_1 v_1 v_2 + \alpha_2 v_1 - \alpha_2 v_2)/(v_1+v_2)^2\right]}{(1+v_2/v_1)^{2L-1}}\frac{1}{2\pi j}\int_\Gamma f(p)\,dp \tag{B–12}$$

where

$$f(p) = \frac{[1+(v_2/v_1)p]^{2L-1}}{p^L(1-p)}\exp\left[\frac{A_2(v_2/v_1)}{v_1+v_2}p + \frac{A_3(v_1/v_2)}{v_1+v_2}\frac{1}{p}\right] \tag{B–13}$$

and Γ is a circular contour of radius less than unity that encloses the origin.

The third step is to evaluate the integral

$$\begin{aligned} \frac{1}{2\pi j}\int_\Gamma f(p)\,dp = \frac{1}{2\pi j}\int_\Gamma &\frac{[1+(v_2/v_1)p]^{2L-1}}{p^L(1-p)} \\ &\times \exp\left[\frac{A_2(v_2/v_1)}{v_1+v_2}p + \frac{A_3(v_1/v_2)}{v_1+v_2}\frac{1}{p}\right]dp \end{aligned} \tag{B–14}$$

In order to facilitate subsequent manipulations, the constants $a \geq 0$ and $b \geq 0$ are introduced and defined as follows:

$$\tfrac{1}{2}a^2 = \frac{A_3(v_1/v_2)}{v_1+v_2}, \qquad \tfrac{1}{2}b^2 = \frac{A_2(v_2/v_1)}{v_1+v_2} \tag{B–15}$$

Let us also expand the function $[1+(v_2/v_1)p]^{2L-1}$ as a binomial series. As a result, we obtain

$$\begin{aligned}\frac{1}{2\pi j}\int_\Gamma f(p)\,dp = &\sum_{k=0}^{2L-1}\binom{2L-1}{k}\left(\frac{v_2}{v_1}\right)^k \\ &\times \frac{1}{2\pi j}\int_\Gamma \frac{p^k}{p^L(1-p)}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp\end{aligned} \tag{B–16}$$

The contour integral given in Equation B–16 is one representation of the Bessel function. It can be solved by making use of the relations

$$I_n(ab)=\begin{cases}\dfrac{1}{2\pi j}\left(\dfrac{a}{b}\right)^n\displaystyle\int_\Gamma \frac{1}{p^{n+1}}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp \\ \dfrac{1}{2\pi j}\left(\dfrac{b}{a}\right)^n\displaystyle\int_\Gamma p^{n-1}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp\end{cases}$$

where $I_n(x)$ is the nth-order modified Bessel function of the first kind and the series representation of Marcum's Q function in terms of Bessel functions, i.e.,

$$Q_1(a,b)=\exp\left[-\tfrac{1}{2}(a^2+b^2)\right]+\sum_{n=0}^{\infty}\left(\frac{a}{b}\right)^n I_n(ab)$$

First, consider the case $0\le k\le L-2$ in Equation B–16. In this case, the resulting contour integral can be written in the form[†]

$$\frac{1}{2\pi j}\int_\Gamma \frac{1}{p^{L-K}(1-p)}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp = Q_1(a,b)\exp\left[\tfrac{1}{2}(a^2+b^2)\right]+\sum_{n=1}^{L-1-k}\left(\frac{b}{a}\right)^n I_n(ab) \tag{B–17}$$

Next, consider the term $k=L-1$. The resulting contour integral can be expressed in terms of the Q function as follows:

$$\frac{1}{2\pi j}\int_\Gamma \frac{1}{p(1-p)}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp = Q_1(a,b)\exp\left[\tfrac{1}{2}(a^2+b^2)\right] \tag{B–18}$$

[†] This contour integral is related to the generalized Marcum Q function, defined as

$$Q_m(a,b)=\int_b^\infty x(x/a)^{m-1}\exp\left[-\tfrac{1}{2}(x^2+a^2)\right]I_{m-1}(ax)\,dx,\qquad m\ge 1$$

in the following manner:

$$Q_m(a,b)\exp\left[\tfrac{1}{2}(a^2+b^2)\right]=\frac{1}{2\pi j}\int_\Gamma \frac{1}{p^m(1-p)}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp$$

Finally, consider the case $L \le k \le 2L-1$. We have

$$\begin{aligned}
\frac{1}{2\pi j}&\int_\Gamma \frac{p^{k-L}}{1-p}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp\\
&=\sum_{n=0}^{\infty}\frac{1}{2\pi j}\int_\Gamma p^{k-L+n}\exp\left(\frac{\frac{1}{2}a^2}{p}+\tfrac{1}{2}b^2p\right)dp\\
&=\sum_{n=k+1-L}^{\infty}\left(\frac{a}{b}\right)^n I_n(ab)=Q_1(a,b)\exp\left[\tfrac{1}{2}(a^2+b^2)\right]-\sum_{n=0}^{k-L}\left(\frac{a}{b}\right)^n I_n(ab)
\end{aligned}\tag{B–19}$$

Collecting the terms that are indicated on the right-hand side of Equation B–16 and using the results given in Equations B–17 to B–19, the following expression for the contour integral is obtained after some algebra:

$$\begin{aligned}
\frac{1}{2\pi j}\int_\Gamma f(p)\,dp=&\left(1+\frac{v_2}{v_1}\right)^{2L-1}\left\{\exp\left[\tfrac{1}{2}(a^2+b^2)\right]Q_1(a,b)-I_0(ab)\right\}\\
&+I_0(ab)\sum_{k=0}^{L-1}\binom{2L-1}{k}\left(\frac{v_2}{v_1}\right)^k\\
&+\sum_{n=1}^{L-1}I_n(ab)\sum_{k=0}^{L-1-n}\binom{2L-1}{k}\left[\left(\frac{b}{a}\right)^n\left(\frac{v_2}{v_1}\right)^k-\left(\frac{a}{b}\right)^n\left(\frac{v_2}{v_1}\right)^{2L-1-k}\right]
\end{aligned}\tag{B–20}$$

Equation B–20 in conjunction with Equation B–12 gives the result for the probability of error. A further simplification results when one uses the following identity, which can easily be proved:

$$\exp\left[\frac{v_1v_2}{(v_1+v_2)^2}(-2\alpha_1v_1v_2+\alpha_2v_1-\alpha_2v_2)\right]=\exp\left[-\tfrac{1}{2}(a^2+b^2)\right]$$

Therefore, it follows that

$$\begin{aligned}
P_b=&\,Q_1(a,b)-I_0(ab)\exp\left[-\tfrac{1}{2}(a^2+b^2)\right]\\
&+\frac{I_0(ab)\exp\left[-\frac{1}{2}(a^2+b^2)\right]}{(1+v_2/v_1)^{2L-1}}\sum_{k=0}^{L-1}\binom{2L-1}{k}\left(\frac{v_2}{v_1}\right)^k+\frac{\exp\left[-\frac{1}{2}(a^2+b^2)\right]}{(1+v_2/v_1)^{2L-1}}\\
&\times\sum_{n=1}^{L-1}I_n(ab)\sum_{k=0}^{L-1-n}\binom{2L-1}{k}\\
&\times\left[\left(\frac{b}{a}\right)^n\left(\frac{v_2}{v_1}\right)^k-\left(\frac{a}{b}\right)^n\left(\frac{v_2}{v_1}\right)^{2L-1-k}\right],\qquad L>1\\
P_b=&\,Q_1(a,b)-\frac{v_2/v_1}{1+v_2/v_1}I_0(ab)\exp\left[-\tfrac{1}{2}(a^2+b^2)\right],\qquad L=1
\end{aligned}\tag{B–21}$$

This is the desired expression for the probability of error. It is now a simple matter to relate the parameters a and b to the moments of the pairs $\{X_k, Y_k\}$. Substituting for A_2 and A_3 from Equation B–10 into Equation B–15, we obtain

$$
\begin{aligned}
a &= \left[\frac{2v_1^2 v_2(\alpha_1 v_2 - \alpha_2)}{(v_1 + v_2)^2}\right]^{1/2} \\
b &= \left[\frac{2v_1 v_2^2(\alpha_1 v_1 + \alpha_2)}{(v_1 + v_2)^2}\right]^{1/2}
\end{aligned}
\qquad \text{(B–22)}
$$

Since v_1, v_2, α_1, and α_2 have been given in Equations B–6 and B–8 directly in terms of the moments of the pairs X_k and Y_k, our task is completed.

APPENDIX C

Error Probabilities for Adaptive Reception of M-Phase Signals

In this appendix, we derive probabilities of error for two- and four-phase signaling over an L-diversity-branch time-invariant Gaussian noise channel and for M-phase signaling over an L-diversity-branch Rayleigh fading additive Gaussian noise channel. Both channels corrupt the signaling waveforms transmitted through them by introducing additive white Gaussian noise and an unknown or random multiplicative gain and phase shift in the transmitted signal. The receiver processing consists of cross-correlating the signal plus noise received over each diversity branch by a noisy reference signal, which is derived either from the previously received information-bearing signals or from the transmission and reception of a pilot signal, and adding the outputs from all L-diversity branches to form the decision variable.

C.1
MATHEMATICAL MODEL FOR AN M-PHASE SIGNALING COMMUNICATION SYSTEM

In the general case of M-phase signaling, the signaling waveforms at the transmitter are[†]

$$s_n(t) = \mathrm{Re}\left[s_{ln}(t)e^{j2\pi f_c t}\right]$$

where

$$s_{ln}(t) = g(t)\exp\left[j\frac{2\pi}{M}(n-1)\right], \qquad n = 1, 2, \ldots, M, \qquad 0 \le t \le T \qquad \text{(C–1)}$$

and T is the time duration of the signaling interval.

Consider the case in which one of these M waveforms is transmitted, for the duration of the signaling interval, over L channels. Assume that each of the channels corrupts the signaling waveform transmitted through it by introducing a multiplicative gain and phase shift, represented by the complex-valued number g_k, and an additive noise $z_k(t)$. Thus, when the transmitted waveform is $s_{ln}(t)$, the waveform received over the kth channel is

---

[†] The complex representation of real signals is used throughout. Complex conjugation is denoted by an asterisk.

$$r_{lk}(t) = g_k s_{ln}(t) + z_k(t), \qquad 0 \le t \le T, \qquad k = 1, 2, \ldots, L \tag{C–2}$$

The noises $\{z_k(t)\}$ are assumed to be sample functions of a stationary white Gaussian random process with zero-mean and autocorrelation function $\phi_z(\tau) = N_0\delta(\tau)$, where N_0 is the value of the spectral density. These sample functions are assumed to be mutually statistically independent.

At the demodulator, $r_{lk}(t)$ is passed through a filter whose impulse response is matched to the waveform $g(t)$. The output of this filter, sampled at time $t = T$, is denoted as

$$X_k = 2\mathcal{E} g_k \exp\left[j\frac{2\pi}{M}(n-1)\right] + N_k \tag{C–3}$$

where $\mathcal{E}$ is the transmitted signal energy per channel and N_k is the noise sample from the kth filter. In order for the demodulator to decide which of the M phases was transmitted in the signaling interval $0 \le t \le T$, it attempts to undo the phase shift introduced by each channel. In practice, this is accomplished by multiplying the matched filter output X_k by the complex conjugate of an estimate $\hat{g}_k$ of the channel gain and phase shift. The result is a weighted and phase-shifted sampled output from the kth-channel filter, which is then added to the weighted and phase-shifted sampled outputs from the other $L - 1$ channel filters.

The estimate $\hat{g}_k$ of the gain and phase shift of the kth channel is assumed to be derived either from the transmission of a pilot signal or by undoing the modulation on the information-bearing signals received in previous signaling intervals. As an example of the former, suppose that a pilot signal, denoted by $s_{pk}(t)$, $0 \le t \le T$, is transmitted over the kth channel for the purpose of measuring the channel gain and phase shift. The received waveform is

$$g_k s_{pk}(t) + z_{pk}(t), \qquad 0 \le t \le T$$

where $z_{pk}(t)$ is a sample function of a stationary white Gaussian random process with zero-mean and autocorrelation function $\phi_p(\tau) = N_0\delta(\tau)$. This signal plus noise is passed through a filter matched to $s_{pk}(t)$. The filter output is sampled at time $t = T$ to yield the random variable $X_{pk} = 2\mathcal{E}_p g_k + N_{pk}$, where $\mathcal{E}_p$ is the energy in the pilot signal, which is assumed to be identical for all channels, and N_{pk} is the additive noise sample. An estimate of g_k is obtained by properly normalizing X_{pk}, i.e., $\hat{g}_k = g_k + N_{pk}/2\mathcal{E}_p$.

On the other hand, an estimate of g_k can be obtained from the information-bearing signal as follows. If one knew the information component contained in the matched filter output, then an estimate of g_k could be obtained by properly normalizing this output. For example, the information component in the filter output given in Equation C–3 is $2\mathcal{E} g_k \exp[j(2\pi/M)(n-1)]$, and, hence, the estimate is

$$\hat{g}_k = \frac{X_k}{2\mathcal{E}} \exp\left[-j\frac{2\pi}{M}(n-1)\right] = g_k + \frac{N_k'}{2\mathcal{E}}$$

where $N'_k = N_k \exp[-j(2\pi/M)(n-1)]$ and the PDF of N'_k is identical to the PDF of N_k. An estimate that is obtained from the information-bearing signal in this manner is called a *clairvoyant estimate*. Although a physically realizable receiver does not possess such clairvoyance, it can approximate this estimate by employing a time delay of one signaling interval and by feeding back the estimate of the transmitted phase in the previous signaling interval.

Whether the estimate of g_k is obtained from a pilot signal or from the information-bearing signal, the estimate can be improved by extending the time interval over which it is formed to include several prior signaling intervals in a way that has been described by Price (1962a, b). As a result of extending the measurement interval, the signal-to-noise ratio in the estimate of g_k is increased. In the general case where the estimation interval is the infinite past, the normalized *pilot signal estimate* is

$$\hat{g}_k = g_k + \sum_{i=1}^{\infty} c_i N_{pki} \Big/ 2\mathcal{E}_p \sum_{i=1}^{\infty} c_i \tag{C–4}$$

where c_i is the weighting coefficient on the subestimate of g_k derived from the ith prior signal interval and N_{pki} is the sample of additive Gaussian noise at the output of the filter matched to $s_{pk}(t)$ in the ith prior signaling interval. Similarly, the clairvoyant estimate that is obtained from the information-bearing signal by undoing the modulation over the infinite past is

$$\hat{g}_k = g_k + \sum_{i=1}^{\infty} c_i N_{ki} \Big/ 2\mathcal{E} \sum_{i=1}^{\infty} c_i \tag{C–5}$$

As indicated, the demodulator forms the product between $\hat{g}_k^*$ and X_k and adds this to the products of the other $L-1$ channels. The random variable that results is

$$\begin{aligned} z &= \sum_{k=1}^{L} X_k \hat{g}_k^* = \sum_{k=1}^{L} X_k Y_k^* \\ &= z_r + j z_i \end{aligned} \tag{C–6}$$

where, by definition, $Y_k = \hat{g}_k$, $z_r = \text{Re}(z)$, and $z_i = \text{Im}(z)$. The phase of z is the decision variable. This is simply

$$\theta = \tan^{-1}\left(\frac{z_i}{z_r}\right) = \tan^{-1}\left[\text{Im}\left(\sum_{k=1}^{L} X_k Y_k^*\right) \Big/ \text{Re}\left(\sum_{k=1}^{L} X_k Y_k^*\right)\right] \tag{C–7}$$

C.2

CHARACTERISTIC FUNCTION AND PROBABILITY DENSITY FUNCTION OF THE PHASE θ

The following derivation is based on the assumption that the transmitted signal phase is zero, i.e., $n = 1$. If desired, the PDF of θ conditional on any other transmitted signal phase can be obtained by translating $p(\theta)$ by the angle $2\pi(n-1)/M$. We also assume that the complex-valued numbers $\{g_k\}$, which characterize the L channels, are mutually statistically independent and identically distributed zero-mean Gaussian random variables. This characterization is appropriate for slowly fading Rayleigh channels. As a consequence, the random variables (X_k, Y_k) are correlated, complex-valued, zero-mean, Gaussian, and statistically independent, but identically distributed with any other pair (X_i, Y_i).

The method that has been used in evaluating the probability density $p(\theta)$ in the general case of diversity reception is as follows. First, the characteristic function of the joint probability distribution function of z_r and z_i, where z_r and z_i are two components that make up the decision variable θ, is obtained. Second, the double Fourier transform of the characteristic function is performed and yields the density $p(z_r, z_i)$. Then the transformation

$$r = \sqrt{z_r^2 + z_i^2}, \qquad \theta = \tan^{-1}\left(\frac{z_i}{z_r}\right) \tag{C–8}$$

yields the joint PDF of the envelope r and the phase θ. Finally, integration of this joint PDF over the random variable r yields the PDF of θ.

The joint characteristic function of the random variables z_r and z_i can be expressed in the form

$$\psi(jv_1, jv_2) = \left[\frac{\dfrac{4}{m_{xx}m_{yy}(1-|\mu|^2)}}{\left(v_1 - j\dfrac{2|\mu|\cos\varepsilon}{\sqrt{m_{xx}m_{yy}}(1-|\mu|^2)}\right)^2 + \left(v_2 - j\dfrac{2|\mu|\sin\varepsilon}{\sqrt{m_{xx}m_{yy}}\,(1-|\mu|^2)}\right)^2 + \dfrac{4}{m_{xx}m_{yy}(1-|\mu|^2)^2}} \right] \tag{C–9}$$

where, by definition,

$$\begin{aligned} m_{xx} &= E\left(|X_k|^2\right), &&\text{identical for all } k \\ m_{yy} &= E\left(|Y_k|^2\right), &&\text{identical for all } k \\ m_{xy} &= E\left(X_k Y_k^*\right), &&\text{identical for all } k \\ \mu &= \frac{m_{xy}}{\sqrt{m_{xx}m_{yy}}} = |\mu|e^{-j\varepsilon} \end{aligned} \tag{C–10}$$

The result of Fourier-transforming the function $\psi(jv_1, jv_2)$ with respect to the variables v_1 and v_2 is

$$\begin{aligned} p(z_r, z_i) = {} & \frac{\left(1-|\mu|^2\right)^L}{(L-1)!\pi 2^L}\left(\sqrt{z_r^2 + z_l^2}\right)^{L-1} \\ & \times \exp[|\mu|(z_r\cos\varepsilon + z_i\sin\varepsilon)]K_{L-1}\left(\sqrt{z_r^2 + z_i^2}\right) \end{aligned} \tag{C–11}$$

where $K_n(x)$ is the modified Hankel function of order n. Then the transformation of random variables, as indicated in Equation C–8 yields the joint PDF of the envelope r and the phase θ in the form

$$p(r,\theta)=\frac{\left(1-|\mu|^2\right)^L}{(L-1)!\pi 2^L}r^L\exp[|\mu|r\cos(\theta-\varepsilon)]K_{L-1}(r) \tag{C–12}$$

Now, integration over the variable r yields the marginal PDF of the phase θ. We have evaluated the integral to obtain $p(\theta)$ in the form

$$\begin{aligned}p(\theta)=\frac{(-1)^{L-1}(1-|\mu|^2)^L}{2\pi(L-1)!}&\left\{\frac{\partial^{L-1}}{\partial b^{L-1}}\left[\frac{1}{b-|\mu|^2\cos^2(\theta-\varepsilon)}\right.\right.\\&\left.\left.\left.+\frac{|\mu|\cos(\theta-\varepsilon)}{[b-|\mu|^2\cos^2(\theta-\varepsilon)]^{3/2}}\cos^{-1}\left(-\frac{|\mu|\cos(\theta-\varepsilon)}{b^{1/2}}\right)\right]\right\}\right|_{b=1}\end{aligned} \tag{C–13}$$

In this equation, the notation

$$\left.\frac{\partial^L}{\partial b^L}f(b,\mu)\right|_{b=1}$$

denotes the Lth partial derivative of the function $f(b,\mu)$ evaluated at $b=1$.

C.3

ERROR PROBABILITIES FOR SLOWLY FADING RAYLEIGH CHANNELS

In this section, the probability of a character error and the probability of a binary digit error are derived for M-phase signaling. The probabilities are evaluated via the probability density function and the probability distribution function of θ.

The probability distribution function of the phase In order to evaluate the probability of error, we need to evaluate the definite integral

$$P(\theta_1\le\theta\le\theta_2)=\int_{\theta_1}^{\theta_2}p(\theta)\,d\theta$$

where θ_1 and θ_2 are limits of integration and $p(\theta)$ is given by Equation C–13. All subsequent calculations are made for a real cross-correlation coefficient μ. A real-valued μ implies that the signals have symmetric spectra. This is the usual situation encountered. Since a complex-valued μ causes a shift of ε in the PDF of θ, i.e., ε is simply a bias term, the results that are given for real μ can be altered in a trivial way to cover the more general case of complex-valued μ.

In the integration of $p(\theta)$, only the range $0\le\theta\le\pi$ is considered, because $p(\theta)$ is an even function. Furthermore, the continuity of the integrand and its derivatives and the fact that the limits θ_1 and θ_2 are independent of b allow for the interchange of integration and differentiation. When this is done, the resulting integral can be evaluated quite readily and can be expressed as follows:

$$\int_{\theta_1}^{\theta_2} p(\theta)\,d\theta = \frac{(-1)^{L-1}(1-\mu^2)^L}{2\pi(L-1)!} \times \frac{\partial^{L-1}}{\partial b^{L-1}} \left\{ \frac{1}{b-\mu^2} \left[\frac{\mu\sqrt{1-(b/\mu^2-1)x^2}}{b^{1/2}} \cot^{-1} x - \cot^{-1}\left(\frac{x b^{1/2}/\mu}{\sqrt{1-(b/\mu^2-1)x^2}} \right) \right] \right\}_{x_1}^{x_2} \Bigg|_{b=1} \tag{C–14}$$

where, by definition,

$$x_i = \frac{-\mu\cos\theta_i}{\sqrt{b-\mu^2(\cos\theta_i)^2}}, \qquad i = 1, 2 \tag{C–15}$$

Probability of a symbol error The probability of a symbol error for any M-phase signaling system is

$$P_e = 2\int_{\pi/M}^{\pi} p(\theta)\,d\theta$$

When Equation C–14 is evaluated at these two limits, the result is

$$P_e = \frac{(-1)^{L-1}(1-\mu^2)^L}{\pi(L-1)!} \frac{\partial^{L-1}}{\partial b^{L-1}} \left\{ \frac{1}{b-\mu^2} \left[\frac{\pi}{M}(M-1) - \frac{\mu\sin(\pi/M)}{\sqrt{b-\mu^2\cos^2(\pi/M)}} \cot^{-1}\left(\frac{-\mu\cos(\pi/M)}{\sqrt{b-\mu^2\cos^2(\pi/M)}} \right) \right] \right\} \Bigg|_{b=1} \tag{C–16}$$

Probability of a binary digit error First, let us consider two-phase signaling. In this case, the probability of a binary digit error is obtained by integrating the PDF $p(\theta)$ over the range $\frac{1}{2}\pi < \theta < 3\pi$. Since $p(\theta)$ is an even function and the signals are a priori equally likely, this probability can be written as

$$P_2 = 2\int_{\pi/2}^{\pi} p(\theta)\,d\theta$$

It is easily verified that $\theta_1 = \frac{1}{2}\pi$ implies $x_i = 0$ and $\theta_2 = \pi$ implies $x_2 = \mu/\sqrt{b-\mu^2}$. Thus,

$$P_2 = \frac{(-1)^{L-1}(1-\mu^2)^L}{2(L-1)!} \frac{\partial^{L-1}}{\partial b^{L-1}} \left[\frac{1}{b-\mu^2} - \frac{\mu}{b^{1/2}(b-\mu^2)} \right] \Bigg|_{b=1} \tag{C–17}$$

After performing the differentiation indicated in Equation C–17 and evaluating the resulting function at $b = 1$, the probability of a binary digit error is obtained in the form

$$P_2 = \frac{1}{2}\left[1 - \mu \sum_{k=0}^{L-1} \binom{2k}{k} \left(\frac{1-\mu^2}{4}\right)^k\right] \tag{C–18}$$

Next, we consider the case of four-phase signaling in which a Gray code is used to map pairs of bits into phases. Assuming again that the transmitted signal is $s_{l1}(t)$, it is clear that a single error is committed when the received phase is $\frac{1}{4}\pi < \theta < \frac{3}{4}\pi$, and a double error is committed when the received phase is $\frac{3}{4}\pi < \theta < \pi$. That is, the probability of a binary digit error is

$$P_{4b} = \int_{\pi/4}^{3\pi/4} p(\theta)\, d\theta + 2\int_{3\pi/4}^{\pi} p(\theta)\, d\theta \tag{C–19}$$

It is easily established from Equations C–14 and C–19 that

$$P_{4b} = \frac{(-1)^{L-1}(1-\mu^2)^L}{2(L-1)!} \frac{\partial^{L-1}}{\partial b^{L-1}} \left[\frac{1}{b-\mu^2} - \frac{\mu}{(b-\mu^2)(2b-\mu^2)^{1/2}}\right]\Bigg|_{b=1}$$

Hence, the probability of a binary digit error for four-phase signaling is

$$P_{4b} = \frac{1}{2}\left[1 - \frac{\mu}{\sqrt{2-\mu^2}} \sum_{k=0}^{L-1} \binom{2k}{k} \left(\frac{1+\mu^2}{4-2\mu^2}\right)^k\right] \tag{C–20}$$

Note that if one defines the quantity $\rho = \mu/\sqrt{2-\mu^2}$, the expression for P_{4b} in terms of ρ is

$$P_{4b} = \frac{1}{2}\left[1 - \rho \sum_{k=0}^{L-1} \binom{2k}{k} \left(\frac{1-\rho^2}{4}\right)^k\right] \tag{C–21}$$

In other words, P_{4b} has the same form as P_2 given in Equation C–18. Furthermore, note that ρ, just like μ, can be interpreted as a cross-correlation coefficient, since the range of ρ is $0 \le \rho \le 1$ for $0 \le \mu \le 1$. This simple fact will be used in Section C.4.

The above procedure for obtaining the bit error probability for an M-phase signal with a Gray code can be used to generate results for $M = 8$, 16, etc., as shown by Proakis (1968).

Evaluation of the cross-correlation coefficient The expressions for the probabilities of error given above depend on a single parameter, namely, the cross-correlation coefficient μ. The clairvoyant estimate is given by Equation C–5, and the matched filter output, when signal waveform $s_{l1}(t)$ is transmitted, is $X_k = 2\mathcal{E}g_k + N_k$. Hence, the cross-correlation coefficient is

$$\mu = \frac{\sqrt{\nu}}{\sqrt{\left(\bar{\gamma}_c^{-1} + 1\right)\left(\bar{\gamma}_c^{-1} + \nu\right)}} \tag{C–22}$$

where, by definition,

$$\nu = \left|\sum_{i=1}^{\infty} c_i\right|^2 \Big/ \sum_{i=1}^{\infty} |c_i|^2$$
$$\bar{\gamma}_c = \frac{\mathcal{E}}{N_0} E\left(|g_k|^2\right), \qquad k = 1, 2, \ldots, L \tag{C–23}$$

The parameter ν represents the effective number of signaling intervals over which the estimate is formed, and $\bar{\gamma}_c$ is the average SNR per channel.

In the case of differential phase signaling, the weighting coefficients are $c_1 = 1$, $c_i = 0$ for $i \neq 1$. Hence, $\nu = 1$ and $\mu = \bar{\gamma}_c/(1 + \bar{\gamma})_c)$.

When $\nu = \infty$, the estimate is perfect and

$$\lim_{\nu \to \infty} \mu = \sqrt{\frac{\bar{\gamma}_c}{\bar{\gamma}_c + 1}}$$

Finally, in the case of a pilot signal estimate given by Equation C–4, the cross-correlation coefficient is

$$\mu = \left[\left(1 + \frac{r+1}{r\bar{\gamma}_t}\right)\left(1 + \frac{r+1}{\nu\bar{\gamma}_t}\right)\right]^{-1/2} \tag{C–24}$$

where, by definition.

$$\bar{\gamma}_t = \frac{\mathcal{E}_t}{N_0} E(|g_k|^2)$$
$$\mathcal{E}_t = \mathcal{E} + \mathcal{E}_p$$
$$r = \mathcal{E}/\mathcal{E}_p$$

The values of μ given above are summarized in Table C–1.

TABLE C–1
Rayleigh Fading Channel

Type of estimate	**Cross-correlation coefficient μ**
Clairvoyant estimate	$\dfrac{\sqrt{\nu}}{\sqrt{\left(\bar{\gamma}_c^{-1} + 1\right)\left(\hat{\gamma}_c^{-1} + \nu\right)}}$
Pilot signal estimate	$\dfrac{\sqrt{r\nu}}{(r+1)\sqrt{\left(\frac{1}{\bar{\gamma}_t} + \frac{r}{r+1}\right)\left(\frac{1}{\bar{\gamma}_t} + \frac{\nu}{r+1}\right)}}$
Differential phase signaling	$\dfrac{\bar{\gamma}_c}{\bar{\gamma}_c + 1}$
Perfect estimate	$\sqrt{\dfrac{\bar{\gamma}_c}{\bar{\gamma}_c + 1}}$

C.4

ERROR PROBABILITIES FOR TIME-INVARIANT AND RICEAN FADING CHANNELS

In Section C.2, the complex-valued channel gains $\{g_k\}$ were characterized as zero-mean Gaussian random variables, which is appropriate for Rayleigh fading channels. In this section, the channel gains $\{g_k\}$ are assumed to be nonzero-mean Gaussian random variables. Estimates of the channel gains are formed by the demodulator and are used as described in Section C.1. Moreover, the decision variable θ is defined again by Equation C–7. However, in this case, the Gaussian random variables X_k and Y_k, which denote the matched filter output and the estimate, respectively, for the kth channel, have nonzero-means, which are denoted by $\bar{X}_k$ and $\bar{Y}_k$. Furthermore, the second moments are

$$\begin{aligned} m_{xx} &= E\left(|X_k - \bar{X}_k|^2\right), &&\text{identical for all channels}\\ m_{yy} &= E\left(|Y_k - \bar{Y}_k|^2\right), &&\text{identical for all channels}\\ m_{xy} &= E\left[(X_k - \bar{X}_k)(Y_k^* - \bar{Y}_k^*)\right], &&\text{identical for all channels}\end{aligned}$$

and the normalized covariance is defined as

$$\mu = \frac{m_{xy}}{\sqrt{m_{xx}m_{yy}}}$$

Error probabilities are given below only for two- and four-phase signaling with this channel model. We are interested in the special case in which the fluctuating component of each of the channel gains $\{g_k\}$ is zero, so that the channels are time-invariant. If, in addition to this time invariance, the noises between the estimate and the matched filter output are uncorrelated, then $\mu = 0$.

In the general case, the probability of error for two-phase signaling over L statistically independent channels characterized in the manner described above can be obtained from the results in Appendix B. In its most general form, the expression for the binary error rate is

$$\begin{aligned} P_2 &= Q_1(a,b) - I_0(ab)\exp[-\tfrac{1}{2}(a^2 - b^2)]\\ &\quad + \frac{I_0(ab)\exp[-\tfrac{1}{2}(a^2+b^2)]}{[2/(1-\mu)]^{2L-1}}\sum_{k=0}^{L-1}\binom{2L-1}{k}\left(\frac{1+\mu}{1-\mu}\right)^k\\ &\quad + \frac{\exp[-\tfrac{1}{2}(a^2+b^2)]}{[2/(1-\mu)]^{2L-1}}\\ &\quad \times \sum_{k=1}^{L-1} I_n(ab)\sum_{k=0}^{L-1-n}\binom{2L-1}{k}\left[\left(\frac{b}{a}\right)^n\left(\frac{1+\mu}{1-\mu}\right)^k - \left(\frac{a}{b}\right)^n\left(\frac{1+\mu}{1-\mu}\right)^{2L-1-k}\right] \quad (L \geq 2)\\ P_2 &= Q_1(a,b) - \tfrac{1}{2}(1+\mu)I_0(ab)\exp[-\tfrac{1}{2}(a^2+b^2)] \quad (L = 1) \end{aligned} \tag{C–25}$$

where, by definition,

$$a = \left(\tfrac{1}{2} \sum_{k=1}^{L} \left| \frac{\bar{X}_k}{\sqrt{m_{xx}}} - \frac{\bar{Y}_k}{\sqrt{m_{yy}}} \right|^2 \right)^{1/2}$$

$$b = \left(\tfrac{1}{2} \sum_{k=1}^{L} \left| \frac{\bar{X}_k}{\sqrt{m_{xx}}} + \frac{\bar{Y}_k}{\sqrt{m_{yy}}} \right|^2 \right)^{1/2} \qquad \text{(C–26)}$$

$$Q_1(a, b) = \int_b^{\infty} x \exp[-\tfrac{1}{2}(a^2 + x^2)] I_0(ax)\, dx$$

$I_n(x)$ is the modified Bessel function of the first kind and of order n.

Let us evaluate the constants a and b when the channel is time-invariant, $\mu = 0$, and the channel gain and phase estimates are those given in Section C.1. Recall that when signal $s_1(t)$ is transmitted, the matched filter output is $X_k = 2\mathcal{E}g_k + N_k$. The clairvoyant estimate is given by Equation C–5. Hence, for this estimate, the moments are $\bar{X}_k = 2\mathcal{E}_{gk}$, $\bar{Y}_k = g_k$, $m_{xx} = 4\mathcal{E}N_0$, and $m_{yy} = N_0/\mathcal{E}\nu$, where $\mathcal{E}$ is the signal energy, N_0 is the value of the noise spectral density, and ν is defined in Equation C–23. Substitution of these moments into Equation C–26 results in the following expressions for a and b:

$$a = \sqrt{\tfrac{1}{2}\gamma_b} |\sqrt{\nu} - 1|$$

$$b = \sqrt{\tfrac{1}{2}\gamma_b} |\sqrt{\nu} + 1| \qquad \text{(C–27)}$$

$$\gamma_b = \frac{\mathcal{E}}{N_0} \sum_{k=1}^{L} |g_k|^2$$

This is a result originally derived by Price (1962).

The probability of error for differential phase signaling can be obtained by setting $\nu = 1$ in Equation C–27.

Next, consider a pilot signal estimate. In this case, the estimate is given by Equation C–4 and the matched filter output is again $X_k = 2\mathcal{E}g_k + N_k$. When the moments are calculated and these are substituted into Equation C–26, the following expressions for a and b are obtained:

$$a = \sqrt{\frac{\gamma_t}{2}} \left| \sqrt{\frac{\nu}{r+1}} - \sqrt{\frac{r}{r+1}} \right|$$

$$b = \sqrt{\frac{\gamma_t}{2}} \left(\sqrt{\frac{\nu}{r+1}} + \sqrt{\frac{r}{r+1}} \right) \qquad \text{(C–28)}$$

where

$$\gamma_t = \frac{\mathcal{E}_t}{N_0} \sum_{k=1}^{L} |g_k|^2$$

$$\mathcal{E}_t = \mathcal{E} + \mathcal{E}_p$$

$$r = \mathcal{E}/\mathcal{E}_p$$

Finally, we consider the probability of a binary digit error for four-phase signaling over a time-invariant channel for which the condition $\mu = 0$ obtains. One approach that can be used to derive this error probability is to determine the PDF of θ and then to integrate this over the appropriate range of values of θ. Unfortunately, this approach proves to be intractable mathematically. Instead, a simpler, albeit roundabout, method may be used that involves the Laplace transform. In short, the integral in Equation 14.4–14 of the text that relates the error probability $P_2(\gamma_b)$ in an AWGN channel to the error probability P_2 in a Rayleigh fading channel is a Laplace transform. Since the bit error probabilities P_2 and P_{4b} for a Rayleigh fading channel, given by Equations C–18 and C–21, respectively, have the same form but differ only in the correlation coefficient, it follows that the bit error probabilities for the time-invariant channel also have the same form. That is, Equation C–25 with $\mu = 0$ is also the expression for the bit error probability of a four-phase signaling system with the parameters a and b modified to reflect the difference in the correlation coefficient. The detailed derivation may be found in the paper by Proakis (1968). The expressions for a and b are given in Table C–2.

TABLE C–2
Time-Invariant Channel

Type of estimate	a	b
Two-phase signaling		
Clairvoyant estimate	$\sqrt{\frac{1}{2}\gamma_b}\lvert\sqrt{\nu}-1\rvert$	$\sqrt{\frac{1}{2}\gamma_b}(\sqrt{\nu}+1)$
Differential phase signaling	0	$\sqrt{2\gamma_b}$
Pilot signal estimate	$\sqrt{\frac{\gamma_t}{2}}\left\lvert\sqrt{\frac{\nu}{r+1}}-\sqrt{\frac{r}{r+1}}\right\rvert$	$\sqrt{\frac{\gamma_t}{2}}\left(\sqrt{\frac{\nu}{r+1}}+\sqrt{\frac{r}{r+1}}\right)$
Four-phase signaling		
Clairvoyant estimate	$\sqrt{\frac{1}{2}\gamma_b}\lvert\sqrt{\nu+1+\sqrt{\nu^2+1}}$ $-\sqrt{\nu+1-\sqrt{\nu^2+1}}\rvert$	$\sqrt{\frac{1}{2}\gamma_b}\Big(\sqrt{\nu+1+\sqrt{\nu^2+1}}$ $+\sqrt{\nu+1-\sqrt{\nu^2+1}}\Big)$
Differential phase signaling	$\sqrt{\frac{1}{2}\gamma_b}\left(\sqrt{2+\sqrt{2}}-\sqrt{2-\sqrt{2}}\right)$	$\sqrt{\frac{1}{2}\gamma_b}\left(\sqrt{2+\sqrt{2}}+\sqrt{2-\sqrt{2}}\right)$
Pilot signal estimate	$\sqrt{\frac{\gamma_t}{4(r+1)}}\Big\lvert\sqrt{\nu+r+\sqrt{\nu^2+r^2}}$ $-\sqrt{\nu+r-\sqrt{\nu^2+r^2}}\Big\rvert$	$\sqrt{\frac{\gamma_t}{4(r+1)}}\Big(\sqrt{\nu+r+\sqrt{\nu^2+r^2}}$ $+\sqrt{\nu+r-\sqrt{\nu^2+r^2}}\Big)$

APPENDIX D

Square Root Factorization

Consider the solution of the set of linear equations

$$\boldsymbol{R}_N \boldsymbol{C}_N = \boldsymbol{U}_N \tag{D–1}$$

where $\boldsymbol{R}_N$ is an $N \times N$ positive-definite symmetric matrix, $\boldsymbol{C}_N$ is an N-dimensional vector of coefficients to be determined, and $\boldsymbol{U}_N$ is an arbitrary N-dimensional vector. The equations in D–1 can be solved efficiently by expressing $\boldsymbol{R}_N$ in the factored form

$$\boldsymbol{R}_N = \boldsymbol{S}_N \boldsymbol{D}_N \boldsymbol{S}_N^t \tag{D–2}$$

where $\boldsymbol{S}_N$ is a lower triangular matrix with elements $\{s_{ik}\}$ and $\boldsymbol{D}_N$ is a diagonal matrix with diagonal elements $\{d_k\}$. The diagonal elements of $\boldsymbol{S}_N$ are set to unity, i.e., $s_{ii} = 1$. Then we have

$$\begin{aligned} r_{ij} &= d\sum_{k=1}^{j} s_{ik} d_k s_{jk}, \qquad 1 \le j \le i-1, \qquad i \ge 2 \\ r_{11} &= d_1 \end{aligned} \tag{D–3}$$

where $\{r_{ij}\}$ are the elements of $\boldsymbol{R}_N$. Consequently, the elements $\{s_{ik}\}$ and $\{d_k\}$ are determined from Equation D–3 according to the equations

$$\begin{aligned} d_1 &= r_{11} \\ s_{ij} d_j &= r_{ij} - \sum_{k=1}^{j-1} s_{ik} d_k s_{jk}, \qquad 1 \le j \le i-1, \qquad 2 \le i \le N \\ d_i &= r_{ii} - \sum_{k=1}^{i-1} s_{ik}^2 d_k, \qquad 2 \le i \le N \end{aligned} \tag{D–4}$$

Thus, Equation D–4 defines $\boldsymbol{S}_N$ and $\boldsymbol{D}_N$ in terms of the elements of $\boldsymbol{R}_N$.

The solution to Equation D–1 is performed in two steps. With Equation D–2 substituted into Equation D–1 we have

$$\boldsymbol{S}_N \boldsymbol{D}_N \boldsymbol{S}_N^t \boldsymbol{C}_N = \boldsymbol{U}_N$$

Let

$$\boldsymbol{Y}_N = \boldsymbol{D}_N \boldsymbol{S}_N^t \boldsymbol{C}_N \tag{D–5}$$

Then

$$\boldsymbol{S}_N \boldsymbol{Y}_N = \boldsymbol{U}_N \tag{D–6}$$

First we solve Equation D–6 for $\boldsymbol{Y}_N$. Because of the triangular form of $\boldsymbol{S}_N$, we have

$$\begin{aligned} y_1 &= u_1 \\ y_i &= u_i - \sum_{j=1}^{i-1} s_{ij} y_j, \qquad 2 \le i \le N \end{aligned} \tag{D–7}$$

Having obtained $\boldsymbol{Y}_N$, the second step is to compute $\boldsymbol{C}_N$. That is,

$$\begin{aligned} \boldsymbol{D}_N \boldsymbol{S}_N^t \boldsymbol{C}_N &= \boldsymbol{Y}_N \\ \boldsymbol{S}_N^t \boldsymbol{C}_N &= \boldsymbol{D}_N^{-1} \boldsymbol{Y}_N \end{aligned}$$

Beginning with

$$c_N = y_N / d_N \tag{D–8}$$

the remaining coefficients of $\boldsymbol{C}_N$ are obtained recursively as follows:

$$c_i = \frac{y_i}{d_i} - \sum_{j=i+1}^{N} s_{ji} c_j, \qquad 1 \le i \le N-1 \tag{D–9}$$

The number of multiplications and divisions required to perform the factorization of $\boldsymbol{R}_N$ is proportional to N^3. The number of multiplications and divisions required to compute $\boldsymbol{C}_N$, once $\boldsymbol{S}_N$ is determined, is proportional to N^2. In contrast, when $\boldsymbol{R}_N$ is Toeplitz the Levinson–Durbin algorithm should be used to determine the solution of Equation D–1, since the number of multiplications and divisions is proportional to N^2. On the other hand, in a recursive least-squares formulation, $\boldsymbol{S}_N$ and $\boldsymbol{D}_N$ are not computed as in Equation D–3, but they are updated recursively. The update is accomplished with N^2 operations (multiplications and divisions). Then the solution for the vector $\boldsymbol{C}_N$ follows the steps of Equations D–5 to D–9. Consequently, the computational burden of the recursive least-squares formulation is proportional to N^2.

References and Bibliography

Abdulrahman, A., Falconer, D. D., and Sheikh, A. U. (1994). "Decision Feedback Equalization for CDMA in Indoor Wireless Communications," *IEEE J. Select. Areas Commun.*, vol. 12, pp. 698–706, May.

Abend, K., and Fritchman, B. D. (1970). "Statistical Detection for Communication Channels with Intersymbol Interference," *Proc. IEEE*, pp. 779–785, May.

Abou-Faycal, I., Trott, M., and Shamai, S. (2001). "The Capacity of Discrete-Time Memoryless Rayleigh-Fading Channels," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1290–1301.

Abramson, N. (1963). *Information Theory and Coding*, McGraw-Hill, New York.

Abramson, N. (1970). "The ALOHA System—Another Alternative for Computer Communications," *1970 Fall Joint Comput. Conf., AFIDS Conf. Proc.*, vol. 37, pp. 281–285, AFIPS Press, Montvale, N.J.

Abramson, N. (1977). "The Throughput of Packet Broadcasting Channels," *IEEE Trans. Commun*, vol. COM-25, pp. 117–128, January.

Abramson, N. (1993). *Multiple Access Communications*, IEEE Press, New York.

Abramson, N. (1994). "Multiple Access in Wireless Digital Networks," *Proc. IEEE*, vol. 82, pp. 1360–1369, September.

Alamouti, A. (1998). "A Simple Transmitter Diversity Scheme for Wireless Communications," *IEEE J. Selected Areas Commun.*, vol. JSAC-16, pp. 1451–1458, October.

Alexander, P. D., Reed, M. C., Asenstorfer, J. A., and Schlegel, C. B. (1999). "Iterative Multiuser Interference Reduction: Turbo CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1008–1014, July.

Al-Hussaini, E., and Al-Bassiouni, A. A. M. (1985). "Performance of MRC Diversity Systems for the Detection of Signals with Nakagami Fading," *IEEE Trans. Commun*, vol. COM-33, pp. 1315–1319, December.

Alouini, M., and Goldsmith, A. (1998). "A Unified Approach for Calculating Error Rates of Linearly Modulated Signals over Generalized Fading Channels," *Proc. IEEE ICC'98*, pp. 459–464, Atlanta, GA.

Altekar, S. A., and Beaulieu, N. C. (1993). "Upper Bounds on the Error Probability of Decision Feedback Equalization," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 145–156, January.

Amihood, P., Milstein, L. B., and Proakis, J. G. (2006). "Analysis of a MISO Pre-BLAST-DFE Technique for Decentralized Receivers," *Proc. Asilomar Conf.*, Pacific Grove, CA, November.

Amihood, P., Masry, E., Milstein, L. B., and Proakis, J. G. (2007). "Performance Analysis of a Pre-BLAST-DFE Technique for MISO Channels with Decentralized Receivers," *IEEE Trans. Commun.,* vol. 55, pp. 1385–1396, July.

Anderson, J. B., Aulin, T., and Sundberg, C. W. (1986). *Digital Phase Modulation*, Plenum, New York.

Anderson, R. R., and Salz, J. (1965). "Spectra of Digital FM," *Bell Syst. Tech. J.,* vol. 44, pp. 1165–1189, July–August.

Annamalai, A., Tellambura, C., and Bhargara, V. K. (1999). "A Unified Approach to Performance Evaluation of Diversity Systems on Fading Channels," in *Wireless Multimedia Network Technologies*, chap. 17, R. Ganesh ed., Kluwer Academic Publishers, Boston, MA.

Annamalai, A., Tellambura, C., and Bhargara, V. K. (1998). "A Unified Analysis of MPSK and MDPSK with Diversity Reception in Different Fading Environments," *IEEE Electr. Lett.*, vol. 34, pp. 1564–1565, August.

Ash, R. B. (1965). *Information Theory*, Interscience, New York.

Aulin, T. (1980). "Viterbi Detection of Continuous Phase Modulated Signals," *Nat Telecommun. Conf. Record*, pp. 14.2.1–14.2.7, Houston, TX, November.

Aulin, T., Rydbeck, N., and Sundberg, C. W. (1981). "'Continuous Phase Modulation—Part II: Partial Response Signaling," *IEEE Trans. Commun.*, vol. COM-29, pp. 210–225, March.

Aulin, T., Sundberg, C. W., and Svensson, A. (1981). "Viterbi Detectors with Reduced Complexity for Partial Response Continuous Phase Modulation," *Conf. Record NTC'81*, pp. A7.61–A7.6.7, New Orleans, LA.

Aulin, T., and Sundberg, C. W. (1981). "Continuous Phase Modulation—Part I: Full Response Signaling," *IEEE Trans. Commun*, vol. COM-29, pp. 196–209, March.

Aulin, T., and Sundberg, C. W. (1982a). "On the Minimum Euclidean Distance for a Class of Signal Space Codes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 43–55, January.

Aulin, T., and Sundberg, C. W. (1982b). "Minimum Euclidean Distance and Power Spectrum for a Class of Smoothed Phase Modulation Codes with Constant Envelope," *IEEE Trans. Commun.*, vol. COM-30, pp. 1721–1729, July.

Aulin, T., and Sundberg, C. W. (1984). "CPM—An Efficient Constant Amplitude Modulation Scheme," *Int. J. Satellite Commun*, vol. 2, pp. 161–186.

Austin, M. E. (1967). "Decision-Feedback Equalization for Digital Communication Over Dispersive Channels," MIT Lincoln Laboratory, Lexington, MA. Tech. Report No. 437, August.

Bahl, L. R., Cocke, J., Jelinek, F., and Raviv, J. (1974). "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate" *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, March.

Barrow, B. (1963). "Diversity Combining of Fading Signals with Unequal Mean Strengths," *IEEE Trans. Commun. Syst.*, vol. CS-11, pp. 73–78, March.

Bauch, G., and Franz, V. (1998). "Iterative Equalization and Decoding for the GSM System," *Proc. VTC '98*, pp. 2262–2266, April.

Bauch, G., Khorram, H., and Hagenauer, J. (1997). "Iterative Equalization and Decoding in Mobile Communications Systems," *Proc. European Personal Mobile Commun. Conf. (EPMCC'77)*, pp. 307–312, September.

Beare, C. T. (1978). "The Choice of the Desired Impulse Response in Combined Linear-Viterbi Algorithm Equalizers," *IEEE Trans. Commun.*, vol. 26, pp. 1301–1307, August.

Beaulieu, N. C. (1990). "An Infinite Series for the Computation of the Complementary Probability Distribution Function of a Sum of Independent Random Variables and Its Application to the Sum of Rayleigh Random Variables," *IEEE Trans. Commun.*, vol. COM-38, pp. 1463–1474, September.

Beaulieu, N. C. (1994). "Bounds on Recovery Times of Decision Feedback Equalizers," *IEEE Trans. Commun*, vol. 42, pp. 2786–2794, October.

Beaulieu, N. C., and Abu-Dayya, A. A. (1991). "Analysis of Equal Gain Diversity on Nakagami Fading Channels," *IEEE Trans. Commun*, vol. COM-39, pp. 225–234, February.

Bégin, G., and Haccoun, D. (1989). "High-Rate Punctured Convolutional Codes: Structure, Properties and Construction Technique," *IEEE Trans. Commun.*, vol. 37, pp. 1381–1385, December.

Bégin, G., Haccoun, D., and Paguin, C. (1990). "Further Results on High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding," *IEEE Trans. Commun.*, vol. 38, pp. 1922–1928, November.

Bekir, N. E., Scholtz, R. A., and Welch, L. R. (1978). "Partial-Period Correlation Properties of PN Sequences," *1978 Nat. Telecommun. Conf. Record*, pp. 35.1.1–25.1.4, Birmingham, Alabama, November.

Belfiore, C. A., and Park, J. H., Jr. (1979). "Decision-Feedback Equalization," *Proc. IEEE*, vol. 67, pp. 1143–1156, August.

Bellini, J. (1986). "Bussgang Techniques for Blind Equalization," *Proc. GLOBECOM'86*, pp. 46.1.1–46.1.7, Houston, TX, December.

Bello, P. (1963). "Characterization of Randomly Time-Variant Linear Channels," *IEEE Trans. Commun.,* vol. 11, pp. 360–393, December.

Bello, P. A., and Nelin, B. D. (1962a). "Predetection Diversity Combining with Selectivity Fading Channels," *IRE Trans. Commun Syst.*, vol. CS-10, pp. 32–42, March.

Bello, P. A., and Nelin, B. D. (1962b). "The Influence of Fading Spectrum on the Binary Error Probabilities of Incoherent and Differentially Coherent Matched Filter Receivers," *IRE Trans. Commun. Syst.*, vol. CS-10, pp. 160–168, June.

Bello, P. A., and Nelin, B. D. (1963). "The Effect of Frequency Selective Fading on the Binary Error Probabilities of Incoherent and Differentially Coherent Matched Filter Receivers," *IEEE Trans. Commun. Syst.*, vol. CS-11, pp. 170–186, June.

Benedetto, S., Ajmone Marsan, M., Albertengo, G., and Giachin, E. (1988). "Combined Coding and Modulation: Theory and Applications," *IEEE Trans. Inform. Theory*, vol. 34, pp. 223–236, March.

Benedetto, S., Divsalar, D., Montorsi, G., and Pollara, F. (1998). "Serial Concatenation of Interleaved Codes: Performance Analysis, Design and Iterative Decoding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 909–926, May.

Benedetto, S., Mondin, M., and Montorsi, G. (1994). "Performance Evaluation of Trellis-Coded Modulation Schemes," *Proc. IEEE*, vol. 82, pp. 833–855, June.

Benedetto, S., and Montorsi, G. (1996). "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 409–428, March.

Bennett, W. R., and Davey, J. R. (1965). *Data Transmission*, McGraw-Hill, New York.

Bennett, W. R., and Rice, S. O. (1963). "Spectral Density and Autocorrelation Functions Associated with Binary Frequency-Shift Keying," *Bell Syst. Tech. J.,* vol. 42, pp. 2355–2385, September.

Bensley, S. E., and Aazhang, B. (1996). "Subspace-Based Channel Estimation for Code-Division Multiple Access Communication Systems," *IEEE Trans. Commun.*, vol. 44, pp. 1009–1020, August.

Benveniste, A., and Goursat, M. (1984). "Blind Equalizers," *IEEE Trans. Commun.*, vol. COM-32, pp. 871–883, August.

Berger, T. (1971). *Rate Distortion Theory*, Prentice-Hall, Englewood Cliffs, NJ.

Berger, T., and Gibson, J. D. (1998). "Lossy Source Coding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2693–2723, October.

Berger, T., and Tufts, D. W. (1967). "Optimum Pulse Amplitude Modulation, Part I: Transmitter-Receiver Design and Bounds from Information Theory," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 196–208.

Bergmans, J. W. M. (1995). "Efficiency of Data-Aided Timing Recovery Techniques," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1397–1408, September.

Bergmans, J. W. M., Rajput, S. A., and Van DeLaar, F. A. M. (1987). "On the Use of Decision Feedback for Simplifying the Viterbi Detector," *Philips J. Research*, vol. 42, no. 4, pp. 399–428.

Bergmans, P. P., and Cover, T. M. (1974). "Cooperative Broadcasting," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 317–324, May.

Berlekamp, E. R. (1968). *Algebraic Coding Theory*, McGraw-Hill, New York.

Berlekamp, E. R. (1973). "Goppa Codes," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 590–592.

Berlekamp, E. R. (1974). *Key Papers in the Development of Coding Theory*, IEEE Press, New York.

Berrou, C., and Glavieux, A. (1996). "Near Optimum Error-Correcting Coding and Decoding: Turbo Codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271.

Berrou, C., Glavieux, A., and Thitimajshima, P. (1993). "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *Proc. IEEE Int. Conf. Commun.*, pp. 1064–1070, May, Geneva, Switzerland.

Bierman, G. J. (1977). *Factorization Methods for Discrete Sequential Estimation*, Academic, New York.

Biglieri, E. (2005). *Coding for Wireless Channels*, Springer, New York.

Biglieri, E., Caire, G., and Taricco, G. (1995). "Approximating the Pairwise Error Probability for Fading Channels," *Electronics Lett.*, vol. 31, pp. 1625–1627.

Biglieri, E., Caire, G., and Taricco, G. (1998a). "Computing Error Probabilities over Fading Channels: A Unified Approach," *European Trans. Telecomm.*, vol. 9, pp. 15–25.

Biglieri, E., Caire, G., Taricco, G., and Ventura-Traveset, J. (1996). "Simple Method for Evaluating Error Probabilities," *Electronics Lett.*, vol. 32, pp. 191–192.

Biglieri, E., Divsalar, D., McLane, P. J., and Simon, M. K. (1991). *Introduction to Trellis-Coded Modulation with Applications*, Macmillan, New York.

Biglieri, E., Proakis, J. G., and Shamai, S. (1998). "Fading Channels: Information-Theoretic and Communications Aspects," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2619–2692, October.

Bingham, J. A. C. (1990). "Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come," *IEEE Commun. Mag.,* vol. 28, pp. 5–14, May.

Bingham J. A. C. (2000). *ADSL, VDSL, and Multicarrier Modulation*, Wiley, New york.

Bjerke, B. A., and Proakis, J. G. (1999). "Multiple Antenna Diversity Techniques for Transmission over Fading Channels," *Proc. WCNC'99*, September, New Orleans, LA.

Blahut, R. E. (1983). *Theory and Practice of Error Control Codes*, Addison-Wesley, Reading, MA.

Blahut, R. E. (1987). *Principles and Practice of Information Theory*, Addison-Wesley, Reading, MA.

Blahut, R. E. (1990). *Digital Transmission of Information*, Addison-Wesley, Reading, MA.

Blahut, R. E. (2003). *Algebraic Codes for Data Transmission*, Cambridge University Press, Cambridge, U.K.

Bose, R. C., and Ray-Chaudhuri, D. K. (1960a). "On a Class of Error Correcting Binary Group Codes," *Inform, Control*, vol. 3, pp. 68–79, March.

Bose, R. C., and Ray-Chaudhuri, D. K. (1960b). "Further Results in Error Correcting Binary Group Codes,' "*Inform. Control*, vol. 3, pp. 279–290, September.

Bottomley, G. E. (1993). "Optimizing the RAKE Receiver for the CDMA Downlink," *Proc. IEEE Veh. Technol. Conf.*, pp. 742–745, Secaucus, N.J.

Bottomley, G. E. Ottosson, T., and Wang, Y. P. E. (2000). "A Generalized RAKE Receiver for Interference Suppression," *IEEE J. Selected Areas Commun.*, vol. 18, pp. 1536–1545, August.

Boutros, J., and Viterbo, E. (1998). "Signal Space Diversity: A Power- and Bandwidth-Efficient Diversity Technique for the Rayleigh Fading Channel," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1453–1467.

Boutros, J., Viterbo, E., Rastello, C., and Belfiore, J.-C. (1996). "Good Lattice Constellations for Both Rayleigh Fading and Gaussian Channels," *IEEE Trans. Inform. Theory*, vol. 42, pp. 502–518.

Boyd, S. (1986). "Multitone Signals with Low Crest Factor," *IEEE Trans. Circuits and Systems*, vol. CAS-33, pp. 1018–1022.

Brennan, D. G. (1959). "Linear Diversity Combining Techniques," *Proc. IRE.*, vol. 47, pp. 1075–1102.

Bucher, E. A. (1980). "Coding Options for Efficient Communications on Non-Stationary Channels," *Rec. IEEE Int. Conf. Commun.*, pp. 4.1.1–4.1.7.

Buehrer, R. M., and Kumar, N. A. (2002) "The Impact of Channel Estimation Error on Space-Time Block Codes," *Proc. IEEE VTC Fall 2002*, pp. 1921–1925, September.

Buehrer, R. M., Nicoloso, S. P., and Gollamudi, S. (1999). "Linear versus Nonlinear Interference Cancellation," *J. Commun. and Networks*, vol. 1, pp. 118–133, June.

Buehrer, R. M., and Woerner, B. D. (1996). "Analysis of Multistage Interference Cancellation for CDMA Using an Improved Gaussian Approximation," *IEEE Trans. Commun.*, vol. 44, pp. 1308–1316, October.

Burton, H. O. (1969). "A Class of Asymptotically Optimal Burst Correcting Block Codes," *Proc. ICCC*, Boulder, CO, June.

Bussgang, J. J. (1952). "Crosscorrelation Functions of Amplitude-Distorted Gaussian Signals," MIT RLE Tech. Report 216.

Cahn, C. R. (1960). "Combined Digital Phase and Amplitude Modulation Communication Systems," *IRE Trans. Common. Syst.*, vol. CS-8, pp. 150–155, September.

Cain, J. B., Clark, G. C., and Geist, J. M. (1979). "Punctured Convolutional Codes of Rate $(n-1)/n$ and Simplified Maximum Likelihood Decoding," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 97–100, January.

Caire, G., and Shamai, S. (1999). "On the Capacity of Some Channels with Channel State Information," *IEEE Trans. Inform. Theory*, vol. 45, pp. 2007–2019.

Caire, G., and Shamai, S. (2003). "On the Achievable Throughput of a Multiantenna Gaussian Broadcast Channel," *IEEE Trans. Inform. Theory,* vol. 43, pp.1691–1706, July.

Caire, G., Taricco, G., and Biglieri, E. (1998). "Bit-Interleaved Coded Modulation," *IEEE Trans. Inform. Theory*, vol. 44, pp. 927–946, May.

Calderbank, A. R. (1998). "The Art of Signalling: Fifty Years of Coding Theory," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2561–2595, October.

Calderbank, A. R., and Sloane, N. J. A. (1987). "New Trellis Codes Based on Lattices and Cosets," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 177–195. March.

Campanella, S. J., and Robinson, G. S. (1971). "A Comparison of Orthogonal Transformations for Digital Speech Processing," *IEEE Trans. Commun.*, vol. COM-19, pp. 1045–1049, December.

Campopiano, C. N., and Glazer, B. G. (1962). "A Coherent Digital Amplitude and Phase Modulation Scheme," *IRE Trans. Commun. Syst.*, vol. CS-10, pp. 90–95, June.

Capetanakis, J. I. (1979). "Tree Algorithms for Packet Broadcast Channels," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 505–515, September.

Caraiscos, C., and Liu, B. (1984). "A Roundoff Error Analysis of the LMS Adaptive Algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 34–41, January.

Carayannis, G., Manolakis, D. G., and Kalouptsidis, N. (1983). "A Fast Sequential Algorithm for Least-Squares Filtering and Prediction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1394–1402, December.

Carayannis, G., Manolakis, D. G., and Kalouptsidis, N. (1986). " A Unified View of Parametric Processing Algorithms for Prewindowed Signals," *Signal Processing*, vol. 10, pp. 335–368, June.

Carleial, A. B., and Hellman, M. E. (1975). "Bistable Behavior of ALOHA-Type Systems," *IEEE Trans. Commun.*, vol. COM-23, pp. 401–410, April 1975.

Carlson, A. B. (1975). *Communication Systems*, McGraw-Hill, New York.

Castagnoli, G., Brauer, S., and Herrmann, M. (1993). "Optimization of Cyclic Redundancy-Check Codes with 24 and 32 Parity Bits," *IEEE Trans. Commun.*, vol. 41, pp. 883–892.

Castagnoli, G., Ganz, J., and Graber, P. (1990). "Optimum Cycle Redundancy-Check Codes with 16-Bit Redundancy," *IEEE Trans. Commun.*, vol. 38, pp. 111–114.

Chang, D. Y., Gersho, A., Ramamurthi, B., and Shohan, Y. (1984). "Fast Search Algorithms for Vector Quantization and Pattern Matching," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, paper 9.11, San Diego, CA, March.

Chang, R. W. (1966). "Synthesis of Band-Limited Orthogonal Signals for Multichannel Data Transmission," *Bell Syst. Tech. J.,* vol. 45, pp. 1775–1796, December.

Chang, R. W. (1971). "A New Equalizer Structure for Fast Start-up Digital Communication," *Bell Syst. Tech. J.*, vol. 50, pp. 1969–2001.

Charash, U. (1979). "Reception Through Nakagami Fading Multipath Channels with Random Delays," *IEEE Trans. Commun.*, vol. COM-27, pp. 657–670, April.

Chase, D. (1972). "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 170–182, January.

Chase, D. (1976). "Digital Signal Design Concepts for a Time-Varying Ricean Channel," *IEEE Trans. Commun.*, vol. COM-24, pp. 164–172, February.

Chen, Z., Zhu, G., and Liu, Y. (2003). "Differential Space-Time Block Codes from Amicable Orthogonal Designs," *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, vol. 2, pp. 768–772, March.

Cherubini, G., Eleftheriou, E., and Olcer, J. (2000). "Filter Bank Modulation Techniques for Very High-Speed Digital Subscriber Lines," *IEEE Commun. Mag.*, pp. 98–104, May.

Cherubini, G., Eleftheriou, E., and Olcer, S. (2002). "Filtered Multitone Modulation for Very High-Speed Digital Subscriber Lines," *IEEE J. Selected Areas Commun.*, vol. 20, pp. 1016–1028, June.

Chevillat, P. R., and Eleftheriou, E. (1989). "Decoding of Trellis-Encoded Signals in the Presence of Intersymbol Interference and Noise," *IEEE Trans. Commun.*, vol. 37, pp. 669–676, July.

Chevillat, P. R., and Eleftheriou, E. (1988). "Decoding of Trellis-Coded Signals in the Presence of Intersymbol Interference and Noise," *Conf. Rec. ICC'88*, pp. 23.1.1–23.1.6, June, Philadelphia, PA.

Chien, R. T. (1964). "Cyclic Decoding Procedures for BCH Codes," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 357–363, October.

Chow, J. S., Tu, J. C., and Cioffi, J. M. (1991). "A Discrete Multitone Transceiver System for HDSL Applications," *IEEE J. Selected Areas Commun.*, vol. SAC-9, pp. 895–908, August.

Chow, J. S., Cioffi, J. M., and Bingham, J. A. C. (1995). "A Practical Discrete Multitone Transceiver Loading Algorithm for Data Transmission over Spectrally Shaped Channels," *IEEE Trans. Commun.*, vol. 43, pp. 773–775, February/March/April.

Chung, S.-Y., Forney, G. D. Jr., Richardson, T., and Urbanke, R. (2001). "On the Design of Low-Density Parity-Check Codes within 0.0045 dB of the Shannon Limit," *IEEE Commun. Lett.*, vol. 5, pp. 58–60.

Chyi, G. T., Proakis, J. G., and Keller, C. M. (1988). "Diversity Selection/Combining Schemes with Excess Noise-Only Diversity Reception Over a Rayleigh-Fading Multipath Channel." *Proc. Conf. Inform. Sci. Syst.*, Princeton University, Princeton, N.J., March.

Ciavaccini, E., and Vitetta, G. M. (2000). "Error Performance of OFDM Signaling over Doubly-Selective Rayleigh Fading Channels," *IEEE Commun., Lett.*, vol. 4 pp. 328–330, November.

Cioffi, J. M., and Kailath, T. (1984a). "Fast Recursive-Least Squares Transversal Filters for Adaptive Filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 304–337, April.

Cioffi, J. M., and Kailath, T. (1984b). "An Efficient Exact-Least-Squares Fractionally Spaced Equalizer Using Intersymbol Interpolation," *IEEE J. Selected Areas Commun.*, vol. 2, pp. 743–756, September.

Clark, A. P., Abdullah, S. N., Jayasinghe, S. J., and Sun, K. H. (1985). "Pseudobinary and Pseudoquaternary Detection Processes for Linearly Distorted Multilevel QAM Signals," *IEEE Trans. Commun.*, vol. COM-33, pp. 639–645, July.

Clark, A. P., and Clayden, M. (1984). "Pseudobinary Viterbi Detector," *Proc. IEE*, vol. 131, part F, pp. 280–218, April.

Cook, C. E., Ellersick, F. W., Milstien, L. B., and Schilling, D. L. (1983). *Spread Spectrum Communications*, IEEE Press, New York.

Costa, M. (1983). "Writing on Dirty Paper," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 439–441, May.

Costas, J. P. (1956). "Synchronous Communications," *Proc. IRE*, vol. 44, pp. 1713–1718, December.

Costello, D. J., Jr., Hagenauer, J., Imai, H., and Wicker, S. B. (1998). "Applications of Error-Control Coding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2531–2560, October.

Cover, T. M. (1972). "Broadcast Channels," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 2–14, January.

Cover, T. M. (1998). "Comments on Broadcast Channels," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2524–2530, October.

Cover, T., and Chiang, M. (2002). "Duality between Channel Capacity and Rate Distortion with Two-Sided State Information," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1629–1638.

Cover, T. M., and Thomas, J. (2006). *Elements of Information Theory*, 2d ed., Wiley, New York.

Cramér, H. (1946). *Mathematical Methods of Statistics*, Princeton University Press, Princeton, NJ.

Damen, O., Chkeif, A., and Belfiore, J. (2000). "Lattice Code Decoder for Space-Time Codes," *IEEE Comm. Lett.*, vol. 4, pp. 161–163, May.

Daneshgaran, F., and Mondin, M. (1999). "Design of Interleavers for turbo codes: Iterative Interleaver Growth Algorithms of Polynomial Complexity," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1845–1859, September.

Daut, D. G., Modestino, J. W., and Wismer, L. D. (1982). "'New Short Constraint Length Convolutional Code Construction for Selected Rational Rates," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 793–799, September.

Davenport, W. B., Jr. (1970). *Probability and Random Processes*, McGraw-Hill, New York.

Davenport, W. B., Jr., and Root, W. L. (1958). *Random Signals and Noise*, McGraw-Hill, New York.

Davisson, L. D. (1973). "Universal Noiseless Coding," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 783–795.

Davisson, L. D., McEliece, R. J., Pursley, M. B., and Wallace, M. S. (1981). "Efficient Universal Noiseless Source codes," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 269–279.

deBuda, R. (1972). "Coherent Demodulation of Frequency Shift Keying with Low Deviation Ratio," *IEEE Trans. Commun.*, vol. COM-20, pp. 429–435, June.

deJong, Y. L. C., and Willink, T. J. (2002). "Iterative Tree Search Detection for MIMO Wireless Systems," *Proc. VTC 2002,* vol. 2, pp. 1041–1045, Vancouver, B. C., Canada, Sept. 24–28.

Deller, J. P., Proakis, J. G., and Hansen, H. L. (2000). *Discrete-Time Processing of Speech Signals*, IEEE Press, New York.

Ding, Z. (1990). *Application Aspects of Blind Adaptive Equalizers in QAM Data Communications*, Ph.D. Thesis, Department of Electrical Engineering, Cornell University.

Ding, Z., Kennedy, R. A., Anderson, B. D. O., and Johnson, C. R. (1989). "Existence and Avoidance of Ill-Convergence of Godard Blind Equalizers in Data Communication Systems," *Proc. 23rd Conf. on Inform. Sci. Systems.*, Baltimore, MD.

Divsalar, D., and Simon, M. (1988a). "The Design of Trellis Coded MPSK for Fading Channels: Performance Criteria," *IEEE Trans. Commun.*, vol. 36, pp. 1004–1012.

Divsalar, D., and Simon, M. (1988b). "The Design of Trellis Coded MPSK for Fading Channels: Set Partitioning for Optimum Code Design," *IEEE Trans. Commun.*, vol. 36, pp. 1013–1021.

Divsalar, D., and Simon, M. K. (1988c). "Multiple Trellis Coded Modulation (MTCM)," *IEEE Trans. Commun.*, vol. COM-36, pp. 410–419.

Divsalar, D., Simon, M. K., and Raphelli, D. (1998). "Improved Parallel Interference Cancellation," *IEEE Trans. Commun.*, vol. 46, pp. 258–268, February.

Divsalar, D., Simon, M. K., and Yuen, J. H. (1987). "Trellis Coding with Asymmetric Modulation," *IEEE Trans. Commun.*, vol. COM-35, pp. 130–141, February.

Divsalar, D., and Yuen, J. H. (1984). "Asymmetric MPSK for Trellis Codes," *Proc. GLOBECOM'84*, pp. 20.6.1–20.6.8, Atlanta, GA, November.

Dixon, R. C. (1976). *Spread Spectrum Techniques*, IEEE Press, New York.

Dobrushin, R. L., and Lupanova, O. B. (1963). *Papers in Information Theory and Cybernetics* (in Russian), Edited by Dobrushin and Lupanova, Izd. Inostr. Lit., Moscow.

Doelz, M. L., Heald, E. T., and Martin, D. L. (1957). "Binary Data Transmission Techniques for Linear Systems," *Proc. IRE*, vol. 45, pp. 656–661, May.

Douillard, C., Jézéquel, M., Berrou, C., Picart, A., Didier, P., and Glavieux, A. (1995). "Iterative Correction of Intersymbol Interference: Turbo-equalization," *ETT European Trans. Telecommun.* vol. 6, pp. 507–511, September/October.

Drouilhet, P. R., Jr., and Bernstein, S. L. (1969). "TATS—A Bandspread Modulation-Demodulation System for Multiple Access Tactical Satellite Communication," *1969 IEEE Electronics and Aerospace Systems (EASCON) Conv. Record*, Washington, DC, pp. 126–132, October 27–29.

Du, J., and Vucetic, B. (1990). "New MPSK Trellis Codes for Fading Channels," *Electronics Lett.*, vol. 26, pp. 1267–1269.

Duel-Hallen, A., and Heegard, C. (1989). "Delayed Decision-Feedback Sequence Estimation," *IEEE Trans. Commun.*, vol. 37, pp. 428–436, May.

Duffy, F. P., and Tratcher, T. W. (1971). "Analog Transmission Performance on the Switched Telecommunications Network," *Bell Syst. Tech. J.,* vol. 50, pp. 1311–1347, April.

Duman, T. M., and Salehi, M. (1997). "New Performance Bounds for Turbo Codes," *Proc. GLOBECOM'97*, pp. 634–638, November, Phoenix, AZ.

Duman, T., and Salehi, M. (1999). "The Union Bound for Turbo-Coded Modulation Systems over Fading Channels," *IEEE Trans. Commun.*, vol. 47, pp. 1495–1502.

Durbin, J. (1959). "Efficient Estimation of Parameters in Moving-Average Models," *Biometrika*, vol. 46, parts 1 and 2, pp. 306–316.

Duttweiler, D. L., Mazo, J. E., and Messerschmitt, D. G. (1974). "Error Propagation in Decision-Feedback Equalizers," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 490–497, July.

Edelman, A. (1989). "Eigenvalue and Condition Numbers of Random Matrices," Ph.D. dissertation, M.I.T., May.

Eleftheriou, E., and Falconer, D. D. (1987). "Adapative Equalization Techniques for HF Channels," *IEEE J. Selected Areas Commun.*, vol. SAC-5, pp. 238–247, February.

El Gamal, A., and Cover, T. M. (1980). "Multiple User Information Theory," *Proc. IEEE*, vol. 68, pp. 1466–1483, December.

Elias, P. (1954). "Error-Free Coding," *IRE Trans. Inform. Theory*, vol. IT-4, pp. 29–37, September.

Elias, P. (1955). "Coding for Noisy Channels," *IRE Convention Record*, vol. 3, part 4, pp. 37–46.

Eriksson, J., and Koivunen, V. (2006). "Complex Random Vectors and ICA Models: Identifiability, Uniqueness, and Separability," *IEEE Trans. Inform. Theory*, vol. 52, pp. 1017–1029.

Esposito, R. (1967). "Error Probabilities for the Nakagami Channel," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 145–148, January.

Eyuboglu, M. V. (1988). "Detection of Coded Modulation Signals on Linear, Severely Distorted Channels Using Decision-Feedback Noise Prediction with Interleaving," *IEEE Trans. Commun.*, vol. COM-36, pp. 401–409, April.

Eyuboglu, M. V., and Qureshi, S. U. H. (1989). "Reduced-State Sequence Estimation for Coded Modulation on Intersymbol Interference Channels," *IEEE J. Selected Areas Commun.*, vol. 7, pp. 989–955, August.

Eyuboglu, M. V., Qureshi, S. U., and Chen, M. P. (1988). "Reduced-State Sequence Estimation for Trellis-Coded Modulation on Intersymbol Interference Channels," *Proc. GLOBEROM '88*, pp., November, Hollywood, FL.

Eyuboglu, M. V., and Qureshi, S. U. (1988). "Reduced-State Sequence Estimation with Set Partitioning and Decision Feedback," *IEEE Trans. Commun.* vol. 36, pp. 13–20, January.

Falconer, D. D. (1976). "Jointly Adaptive Equalization and Carrier Recovery in Two-Dimensional Digital Communication Systems," *Bell Syst. Tech. J.*, vol. 55, pp. 317–334, March.

Falconer, D. D., and Ljung, L. (1978). "Application of Fast Kalman Estimation to Adaptive Equalization," *IEEE Trans. Commun.*, vol. COM-26, pp. 1439–1446, October.

Falconer, D. D., and Magee, F. R. (1973). "Adaptive Channel Memory Truncation for Maximum Likelihood Sequence Estimation," *Bell Syst. Tech. J.*, vol. 52, pp. 1541–1562, November.

Falconer, D. D., and Salz, J. (1977). "Optimal Reception of Digital Data Over the Gaussian Channel with Unknown Delay and Phase Jitter," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 117–126, January.

Fano, R. M. (1961). *Transmission of Information*, MIT Press, Cambridge, MA.

Fano, R. M. (1963). " A Heuristic Discussion of Probabilistic Decoding," *IEEE Trans. Inform. Theory*, vol. IT-9, pp. 64–74, April.

Feinstein, A. (1958). *Foundations of Information Theory*, McGraw-Hill, New York.

Fincke, U., and Pohst, M. (1985). "Improved Methods for Calculating Vectors of Short Length in a Lattice, Including a Complexity Analysis," *Math. Comput.*, vol. 44, pp. 463–471, April.

Fire, P. (1959). "A Class of Multiple-Error-Correcting Binary Codes for Non-Independent Errors," Sylvania Report No. RSL-E-32, Sylvania Electronic Defense Laboratory, Mountain view, CA, March.

Fischer, R. F. H. (2002). *Precoding and Signal Shaping for Digital Transmission*, Wiley, New York.

Fischer, R. F. H., and Huber, J. B. (1996). "A New Loading Algorithm for Discrete Multitone Transmission," *Proc. IEEE GLOBECOM'96*, pp. 724–728, November, London.

Fischer, R. F. H., Windpassinger, C., Lampe, A., and Huber, J. B. (2002). "Space-Time Transmission Using Tomlinson-Harashima Precoding," *Proc. 4th Int. ITG Conf. on Source and Channel Coding*, pp. 139–147, Berlin, January.

Forney, G. D., Jr. (1965). "On Decoding BCH Codes," *IEEE Trans. Inform. Theory*, vol. IT-11, pp. 549–557, October.

Forney, G. D., Jr. (1966a). *Concatenated Codes*, MIT Press, Cambridge, MA.

Forney, G. D., Jr. (1966b). "Generalized Minimum Distance Decoding," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 125–131, April.

Forney, G. D., Jr. (1968). "Exponential Error Bounds for Erasure, List, and Decision-Feedback Schemes," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 206–220, March.

Forney, G. D., Jr. (1970). "Coding and Its Application in Space Communications," *IEEE Spectrum*, vol. 7, pp. 47–58.

Forney, G. D., Jr. (1970a). "Coding and Its Application in Space Communications," *IEEE Spectrum*, vol. 7, pp. 47–58, June.

Forney, G. D., Jr. (1970b). "Convolutional Codes I: Algebraic Structure," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 720–738, November.

Forney, G. D., Jr. (1971). "Burst Correcting Codes for the Classic Bursty Channel," *IEEE Trans. Common. Tech.*, vol. COM-19, pp. 772–781. October.

Forney, G. D., Jr. (1972). "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference." *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 363–378, May.

Forney, G. D., Jr. (1974). "Convolutional Codes III: Sequential Decoding," *Inform. Control*, vol. 25, pp. 267–297, July.

Forney, G. D., Jr. (1988). "Coset Codes I: Introduction and Geometrical Classification," *IEEE Trans. Inform. Theory*, vol. IT-34, pp. 671–680, September.

Forney, G. D., Jr. (2000). "Codes on Graphs: Normal Realizations," in *Information Theory, 2000. Proc. IEEE Int. Symp.*, p. 9.

Forney, G. D., Jr. (2001). "Codes on Graphs: Normal Realizations," *IEEE Trans. Inform. Theory*, vol. 47, pp. 520–548.

Forney, G. D., Jr., Gallager, R. G., Lang, G. R., Longstaff, F. M., and Qureshi, S. U. (1984). "Efficient Modulation for Band-Limited Channels," *IEEE J. Selected Areas Commun.*, vol. SAC-2, pp. 632–647, September.

Forney, G. D., Jr., and Ungerboeck, G. (1998). "Modulation and Coding for Linear Gaussian Channels," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2384–2415, October.

Foschini, G. J. (1977). "A Reduced State Variant of Maximum Likelihood Sequence Detection Attaining Optimum Performance for High Signal-to-Noise Ratios," *IEEE Trans. Inform. Theory*, vol. 23, pp. 605–609.

Foschini, G. J. (1984). "Contrasting Performance of Faster-Binary Signaling with QAM," *Bell Syst. Tech. J.*, vol. 63, pp. 1419–1445, October.

Foschini, G. J. (1985). "Equalizing Without Altering or Detecting Data, *Bell Syst. Tech. J.*, vol. 64, pp. 1885–1911, October.

Foschini, G. J. (1996). "Layered Space-Time Architecture for Wireless-Communication in a Fading Environment When Using Multi-element Antennas," *Bell Labs Tech. J.*, pp. 41–59, Autumn.

Foschini, G. J., and Gans, M. J. (1998). "On Limits of Wireless Communications in a Fading Environment When Using Multiple Antennas," *Wireless Personal Commun.* pp. 311–335, June.

Foschini, G. J., Gitlin, R. D., and Weinstein, S. B. (1974). " Optimization of Two-Dimensional Signal Constellations in the Presence of Gaussian Noise," *IEEE Trans. Commun.*, vol. COM-22, pp. 28–38, January.

Foschini, G. J., Golden, G. D., Valenzuela, R. A., and Wolniansky, P. W. (1999). "Simplified Processing for High Spectral Efficiency Wireless Communication Employing Multi-element Arrays," *IEEE J. Selected Areas Commun.*, vol. 17, pp. 1841–1852, November.

Franks, L. E. (1969). *Signal Theory*, Prentice-Hall, Englewood Cliffs, NJ.

Franks, L. E. (1983). "'Carrier and Bit Synchronization in Data Communication—A Tutorial Review," *IEEE Trans. Commun.*, vol. COM-28, pp. 1107–1121, August.

Franks, L. E. (1981). "Synchronization Subsystems: Analysis and Design," in *Digital Communications, Satellite/Earth Station Engineering*, K. Feher (ed.), Prentice-Hall, Englewood Cliffs, NJ.

Franks, L. E. (1980). "Carrier and Bit Synchronization in Data Communication—A Tutorial Review," *IEEE Trans. Commun.*, vol. COM-28, pp. 1107–1120, August.

Fredricsson, S. (1974). "Optimum Transmitting Filter in Digital PAM Systems with a Viterbi Detector," *IEEE Trans. Inform. Theory*, vol. 20, pp. 479–489.

Fredricsson, S. (1975). "Pseudo-Randomness Properties of Binary Shift Register Sequences," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 115–120, January.

Freiman, C. E., and Wyner, A. D. (1964). "Optimum Block Codes for Noiseless Input Restricted Channels," *Inform. Control*, vol. 7, pp. 398–415.

Frenger, P., Orten, P., Ottosson, T., and Svensson, A. (1998). "Multirate Convolutional Codes:, Tech. Report No. 21, Communication Systems Group, Department of Signals and Systems, Chalmers University of Technology, Goteborg, Sweden, April.

Friese, M. (1997). "OFDM Signals with Low Crest Factor," *IEEE Trans. Commun.*, vol. 45, pp. 1338–1344, October.

Gaarder, N. T. (1971). "Signal Design for Fast-Fading Gaussian Channels," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 247–256, May.

Gabor, A. (1967). "'Adaptive Coding for Self Clocking Recording," *IEEE Trans. Electronic Comp*. vol. EC-16, p. 866.

Gallager, R. G. (1960). "Low-Density Parity-Check Codes," Ph.D. thesis, M.I.T., Cambridge, MA.

Gallager, R. G. (1963). *Low-Density Parity-Check Codes*, The M.I.T. Press, Cambridge, MA.

Gallager, R. G. (1965). "Simple Derivation of the Coding Theorem and Some Applications," *IEEE Trans. Inform. Theory*, vol. IT-11, pp. 3–18, January.

Gallager, R. G. (1968). *Information Theory and Reliable Communication*, Wiley, New York.

Gan, Y. H., and Mow, W. H. (2005). "Accelerated Complex Lattice Reduction Algorithms Applied to MIMO Detection," *Proc. 2005 IEEE Global Telecommunications Conf. (GLOBECOM)*, pp. 2953–2957, St. Louis, MO, Nov. 28–Dec. 2.

Gardner, F. M. (1979). *Phaselock Techniques*, Wiley, New York.

Gardner, W. A. (1984). "Learning Characteristics of Stochastic-Gradient Descent Algorithms: A General Study, Analysis, and Critique", *Signal Processing*, vol. 6, pp. 113–133, April.

Garg, V. K., Smolik, K., and Wilkes, J. E. (1997). *Applications of CDMA in Wireless/Personal Communications*, Prentice-Hall, Upper Saddle River, NJ.

Garth, L. M., and Poor, H. V. (1992). "Narrowband Interference Suppression in Impulsive Channels," *IEEE Trans. Aerospace and Electronic Sys.*, vol. 28, pp. 81–89, January.

George, D. A., Bowen, R. R., and Storey, J. R. (1971). "An Adaptive Decision-Feedback Equalizer," *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 281–293, June.

Gersho, A. (1982). "On the Structure of Vector Quantizers," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 157–166, March.

Gersho, A., and Gray, R. M. (1992). *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston.

Gersho, A., and Lawrence, V. B. (1984). "Multidimensional Signal Constellations for Voiceband Data Transmission," *IEEE J. Selected Areas Commun*, vol. SAC-2, pp. 687–702, September.

Gerst, I., and Diamond, J. (1961). "The Elimination of Intersymbol Interference by Input Pulse Shaping," *Proc. IRE*, vol. 53, July.

Ghez, S., Verdu, S., and Schwartz, S. C. (1988). "Stability Properties of Slotted Aloha with Multipacket Reception Capability," *IEEE Trans. Autom. Control*, vol. 33, pp. 640–649, July.

Ghosh, M., and Weber, C. L. (1991). "Maximum Likelihood Blind Equalization," *Proc. 1991 SPIE Conf.*, San Diego, CA, July.

Giannakis, G. B. (1987). "Cumulants: A Powerful Tool in Signal Processing," *Proc. IEEE*, vol. 75, pp. 1333–1334, September.

Giannakis, G. B., and Mendel, J. M. (1989). "Identification of Nonminimum Phase Systems Using Higher-Order Statistics," *IEEE Trans. Acoust., Speech and Signal Processing*, vol. 37, pp. 360–377, March.

Gibson, J. D., Berger, T., Lookabaugh, T., Lindbergh, D., and Baker, R. L. (1998). *Digital Compression for Multimedia: Principles and Standards*, Morgan Kaufmann, San Francisco, CA.

Gilbert, E. N. (1952). "A Comparison of Signaling Alphabets," *Bell Syst. Tech. J.*, vol. 31, pp. 504–522, May.

Gilhousen, K. S., Jacobs, I. M., Podovani, R., Viterbi, A. J., Weaver, L. A., and Wheatley, G. E. III (1991). "On the Capacity of a Cellular CDMA System," *IEEE Trans. Vehicular Tech*, vol. 40, pp. 303–312, May.

Ginis, G., and Cioffi, J. (2002). "Vectored Transmission for Digital Subscriber Line Systems," *IEEE J. Selected Areas Commun.*, vol. 20, pp. 1085–1104, June.

Gitlin, R. D., Meadors, H. C., and Weinstein, S. B. (1982). "The Tap Leakage Algorithm: An Algorithm for the Stable Operation of a Digitally Implemented Fractionally Spaced, Adaptive Equalizer," *Bell Syst. Tech. J.,* vol. 61, pp. 1817–1839, October.

Gitlin, R. D., and Weinstein, S. B. (1979). "On the Required Tap-Weight Precision for Digitally Implemented Mean-Squared Equalizers," *Bell Syst. Tech. J.*, vol. 58, pp. 301–321, February.

Gitlin, R. D., and Weinstein, S. B. (1981). "Fractionally-Spaced Equalization: An Improved Digital Transversal Equalizer," *Bell Syst. Tech. J.*, vol. 60, pp. 275–296, February.

Glave, F. E. (1972). "An Upper Bound on the Probability of Error due to Intersymbol Interference for Correlated Digital Signals," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 356–362, May.

Goblick, T. J., Jr., and Holsinger, J. L. (1967). "Analog Source Digitization: A Comparison of Theory and Practice," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 323–326, April.

Godard, D. N. (1974). "Channel Equalization Using a Kalman Filter for Fast Data Transmission," *IBM J. Res. Dev.*, vol. 18, pp. 267–273, May.

Godard, D. N. (1980). "Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communications Systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 1867–2875, November.

Golay, M. J. E. (1949). "Note on Digital Coding," *Proc. IRE*, vol. 37, p. 657, June.

Gold, R. (1967). "Optimal Binary Sequences for Spread Spectrum Multiplexing," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 619–621, October.

Gold, R. (1968). "Maximal Recursive Sequences with 3-Valued Recursive Cross Correlation Functions," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 154–156, January.

Goldsmith, A. (2005). *Wireless Communications*, Cambridge University Press, Cambridge, U.K.

Goldsmith, A., and Varaiya, P. (1997). "Capacity of Fading Channels with Channel Side Information," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1986–1992.

Goldsmith, A. J., and Varaiya, P. P. (1996). "Capacity, Mutual Information, and Coding for Finite-State Markov Channels," *IEEE Trans. Inform. Theory*, vol. 42, pp. 868–886.

Golomb, S. W. (1967). *Shift Register Sequences*, Holden-Day, San Francisco, CA.

Goppa, V. D. (1970). "New Class of Linear Correcting Codes," *Probl. Peredach. Inform.*, vol. 6, pp. 24–30.

Goppa, V. D. (1971). "Rational Presentation of Codes and (L, g)-codes," *Probl. Peredach. Inform.*, vol. 7, pp. 41–49.

Gray, R. M. (1975). "Sliding Block Source Coding," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 357–368, July.

Gray, R. M. (1990). *Source Coding Theory*, Kluwer Academic Publishers, Boston.

Gray, R. M., and Neuhoff, D. L. (1998). "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2325–2383, October.

Green, P. E., Jr. (1962). "Radar Astronomy Measurement Techniques," MIT Lincoln Laboratory, Lexington, MA, Tech. Report No. 282, December.

Gronemeyer, S. A., and McBride, A. L. (1976). "MSK and Offset QPSK Modulation," *IEEE Trans. Commun*, vol. COM-24, pp. 809–820, August.

Gu, D., and Leung, (2003). "Performance Analysis of Transmit Diversity Schemes with Imperfect Channnel Estimation," *Electronic Lett.*, vol. 39, pp. 402–403, February.

Gupta, S. C. (1975). "Phase-Locked Loops," *Proc. IEEE*, vol. 63, pp. 291–306, February.

Haccoun, D., and Bégin, G. (1989). "High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding," *IEEE Trans. Commun.*, vol. 37, pp. 1113–1125, November.

Hagenauer, J. (1988). "Rate Compatible Punctured Convolutional Codes and Their Applications," *IEEE Trans. Commun.*, vol. 36, pp. 389–400, April.

Hagenauer, J., and Hoeher, P. (1989). "A Viterbi Algorithm with Soft-Decision Outputs and its Applications," *Proc. IEEE GLOBECOM Conf.*, pp. 1680–1686, November, Dallas, TX.

Hagenauer, J., Offer, E., Méasson, C., and Mörz, M. (1999). "Decoding and Equalization with Analog Non-Linear Networks," *European Trans. Telecommun.*, vol. 10, pp. 659–680, November/December.

Hagenauer, J., Offer, E., and Papke, L. (1996). "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Inform. Theory*, vol. IT-42, pp. 429–445, March.

Hagenauer, J., Seshadri, N., and Sundberg, C.-E. (1990). "The Performance of Rate-Compatible Punctured Convolutional Codes for Digital Mobile Radio," *IEEE Trans. Commun.*, vol. 38, pp. 966–980, July.

Hahn, P. M. (1962). "Theoretical Diversity Improvement in Multiple Frequency Shift Keying," *IRE Trans. Commun. Syst.*, vol. CS-10, pp. 177–184, June.

Hamming, R. W. (1950). "'Error Detecting and Error Correcting Codes," *Bell Syst. Tech. J.*, vol. 29, pp. 147–160, April.

Hamming, R. W. (1986). *Coding and Information Theory*, Prentice-Hall, Englewood Cliffs, NJ.

Hancock, J. C., and Lucky, R. W. (1960). "Performance of Combined Amplitude and Phase-Modulated Communication Systems," *IRE Trans. Commun. syst.*, vol. CS-8, pp. 232–237, December.

Harashima, H., and Miyakawa, H. (1972). "Matched-Transmission Technique for Channels with Intersymbol Interference," *IEEE Trans. Commun.*, vol. COM-20, pp. 774–780.

Hartley, R. V. (1928). "Transmission of Information," *Bell Syst. Tech. J.*, vol. 7, p. 535.

Hatzinakos, D., and Nikias, C. L. (1991). "Blind Equalization Using a Tricepstrum-Based Algorithm," *IEEE Trans. Commun*, vol. COM-39, pp. 669–682, May.

Haykin, S. (1996). *Adaptive Filter Theory*, 3rd ed., Prentice-Hall: Upper Saddle River, NJ.

Haykin, S., and Moher, M. (2005). *Modem Wireless Communications*, Prentice-Hall, Upper Saddle River, NJ.

Hecht, M., and Guida, A. (1969). "Delay Modulation," *Proc. IEEE*, vol. 57, pp. 1314–1316, July.

Heegard, C., and Wicker, S. B. (1999). *Turbo Coding*, Kluwer Academic Publishers, Boston, MA.

Heller, J. A. (1968). "Short Constraint Length Convolutional Codes," Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA *Space Program Summary* 37–54, vol. 3, pp. 171–174, December.

Heller, J. A. (1975). "Feedback Decoding of Convolutional Codes," in *Advances in Communication Systems*, vol. 4, A. J. Viterbi (ed.), Academic, New York.

Heller, J. A., and Jacobs, I. M. (1971). "Viterbi Decoding for Satellite and Space Communication," *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 835–848, October.

Helstrom, C. W. (1955). "The Resolution of Signals in White Gaussian Noise," *Proc. IRE*, vol. 43, pp. 1111–11187, September.

Helstrom, C. W. (1968). *Statistical Theory of Signal Detection*, Pergamon, London.

Helstrom, C. W. (1991). *Probability and Stochastic Processes for Engineers*, Macmillan, New York.

Hildebrand, F. B. (1961). *Methods of Applied Mathematics*, Prentice-Hall, Englewood Cliffs, NJ.

Hirosaki, B. (1981). "An Orthogonality Multiplexed QAM System Using the Discrete Fourier Transform," *IEEE Trans. Commun.*, vol. COM-29, pp. 982–989, July.

Hirosaki, B., Hasegawa, S., and Sabato, A. (1986). "Advanced Group-Band Modem Using Orthogonally Multiplexed QAM Techniques," *IEEE Trans. Commun.*, vol. COM-34, pp. 587–592, June.

Ho, E. Y., and Yeh, Y. S. (1970). "A New Approach for Evaluating the Error Probability in the Presence of Intersymbol Interference and Additive Gaussian Noise," *Bell Syst. Tech. J.*, vol. 49, pp. 2249–2265, November.

Hochwald, B. M., and Sweldens, W. (2000). "Differential Unitary Space-Time Modulation," *IEEE Trans. Commun.*, vol. 48, pp. 2041–2052, December.

Hochwald, B. M., and Vishwanath, S. (2002). "Space-time Multiple Access: Linear Growth in the Sum Rate," *Proc. 40th Allerton Conf. Comput., Commun., Control*, Monticello, IL, pp. 387–396, October.

Hochwald, B. M., and ten Brink, S. (2003). "Achieving Near-Capacity on a Multiple-Antenna Channel," *IEEE Trans. Commun.*, vol. 51, pp. 389–399, March.

Hochwald, B. M., Peel, C. B., and Swindlehurst, A. L. (2005). "A Vector-Perturbation Technique for Near-Capacity Multiantenna Multiuser Communication—Part II: Perturbation," *IEEE Trans. Commun.*, vol. 53, pp. 537–544, March.

Hocquenghem, A. (1959). "'Codes Correcteurs d'Erreurs," *Chiffres*, vol. 2, pp. 147–156.

Hole, K. J. (1988). "New Short Constraint Length Rate $(n = 1)/n$ Punctured Convolutional Codes for Soft-Decision Viterbi Decoding," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1079–1081, September.

Holmes, J. K. (1982). *Coherent Spread Spectrum Systems*, Wiley-Interscience, New York.

Holsinger, J. L. (1964). "Digital Communication over Fixed Time-Continuous Channels with Memory, with Special Application to Telephone Channels," MIT Research Lab. of Electronics, Tech. Rep. 430.

Honig, M. L. (1998). "Adaptive Linear Interference Suppression for Packet DS-CDMA," *European Trans. Telecommun. (ETT)*, vol. 9, pp. 173–181, March–April.

Honig, M. L., Madhow, U., and Verdu, S. (1995). "Blind Adaptive Multiuser Detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July.

Horwood, D., and Gagliardi, R. (1975). "Signal Design for Digital Multiple Access Communications," *IEEE Trans. Commun.*, vol. COM-23, pp. 378–383, March.

Hsu, F. M. (1982). "Square-Root Kalman Filtering for High-Speed Data Received over Fading Dispersive HF Channels," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 753–763, September.

Huffman, D. A. (1952). "A Method for the Construction of Minimum Redundancy Codes," *Proc. IRE*, vol. 40, pp. 1098–1101, September.

Hughes, B. L. (2000). "Differential Space-Time Modulation," *IEEE Trans. Inform. Theory*, vol. 46, pp 2567–2578, November.

Hui, J. Y. N. (1984). "Throughput Analysis for Code Division Multiple Accessing of the Spread Spectrum Channel," *IEEE J. Selected Areas Commun.*, vol. SAC-2, pp. 482–486, July.

Im, G. H., and Un, C. K. (1987). "A Reduced Structure of the Passband Fractionally-Spaced Equalizer," *Proc. IEEE*, vol. 75, pp. 847–849, June.

Itakura, F. (1975). "Minimum Prediction Residual Principle Applied to Speech Recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 67–72, February.

Itakura, F., and Saito, S. (1968). "Analysis Synthesis Telephony Based on the Maximum-Likelihood Methods," *Proc. 6th Int. Congr. Acoust.*, Tokyo, Japan, pp. C17–C20.

Jacobs, I. M. (1974). "Practical Applications of Coding," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 305–310, May.

Jafarkhani, H. (2003). "A Noncoherent Detection Scheme for Space-Time Block Codes," in *Communication, Information, and Network Security*, V. Bhargava et al: (eds.), Kluwer Academic Publishers, Boston.

Jafarkhani, H. (2005). *Space-Time Coding*, Cambridge Univeristy Press, Cambridge, U.K.

Jafarkhani, H., and Tarokh, V. (2001). "Multiple Transmit Antenna Differential Detection from Generalized Orthogonal Designs," *IEEE Trans. Inform Theory*, vol. 47, pp. 2626–2631, September.

Jakes, W. C. (1974). *Microwave Mobile Communications*, Wiley, New York.

Jamali, S. H., and Le-Ngoc, T. (1991). "A New 4-State 8 PSK TCM Scheme for Fast Fading, Shadowed Mobile Radio Channels," *IEEE Trans. Veh. Technol.*, pp. 216–222.

Jamali, S. H., and Le-Ngoc, T. (1994). *Coded Modulation Techniques for Fading Channels*, Kluwer Academic Publishers, Boston.

Jelinek, F. (1968). *Probabilistic Information Theory*, McGraw-Hill, New York.

Jelinek, F. (1969). "Fast Sequential Decoding Algorithm Using a Stack," *IBM J. Res. Dev.*, vol. 13, pp. 675–685, November.

Johannesson, R., and Zigangirov, K. S. (1999). *Fundamentals of Convolutional Coding*, IEEE Press, New York.

Johnson, C. R. (1991). "Admissibility in Blind Adaptive Channel Equalization," *IEEE Control Syst. Mag.*, pp. 3–15, January.

Jones, A. E., Wilkinson, T. A., and Barton, S. K. (1994). "Block Coding Scheme for Reduction of Peak-to-Mean Envelope Power Ratio of Multicarrier Transmission Schemes," *Electr. Lett.*, vol. 30, pp. 2098–2099, December.

Jones, S. K., Cavin, R. K., and Reed, W. M. (1982). "Analysis of Error-Gradient Adaptive Linear Equalizers for a Class of Stationary-Dependent Process," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 318–329, March.

Jootar, J., Zeidler, J. R., and Proakis, J. G. (2005). "Performance of Alamouti Space-Time Code in Time-Varying Channel with Noisy Channel Estimates," *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, vol. 1, pp. 498–503, New Orleans, LA, March 13–17.

Jordan, K. L., Jr. (1966). ""The Performance of Sequential Decoding in Conjunction with Efficient Modulation," *IEEE Trans. Commun. Syst.*, vol. CS-14, pp. 283–287, June.

Justesen, J. (1972). ""A Class of Constructive Asymptotically Good Algebraic Codes," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 652–656, September.

Kailath, T. (1960). "Correlation Detection of Signals Perturbed by a Random Channel," *IRE Trans. Inform. Theory*, vol. IT-6, pp. 361–366, June.

Kailath, T. (1961). "Channel Characterization: Time-Variant Dispersive Channels, in *Lectures on Communication System Theory*, chap. 6, E. Baghdady (ed.), McGraw-Hill, New York.

Kalet, I. (1989). "The Multitone Channel," *IEEE Trans. Commun.*, vol. COM-37, pp. 119–124, February.

Kasami, T. (1966). "Weight Distribution Formula for Some Class of Cyclic Codes," Coordinated Science Laboratory, University of Illinois, Urbana, IL, Tech. Report No. R-285, April.

Kawas Kalet, G. (1989). "Simple Coherent Receivers for Partial Response Continuous Phase Modulation," *IEEE J. Selected Areas Commun.*, vol. 7, pp. 1427–1436, December.

Kaye, A. R., and George, D. A. (1970). "Transmission of Multiplexed PAM Signals over Multiple Channel and Diversity Systems," *IEEE Trans. Commun.*, vol. COM-18, pp. 520–525, October.

Kelly, E. J., Reed, I. S., and Root, W. L. (1960). "The Detection of Radar Echoes in Noise, Pt. I." *J. SIAM*, vol. 8, pp. 309–341, September.

Ketchum, J., and Proakis, J. G. (1982). "Adaptive Algorithms for Estimating and Suppressing Narrowband Interference in PN Spread Spectrum Systems," *IEEE Trans. Commun.*, vol. COM-30, pp. 913–924, May.

Klein, A. (1997). "Data Detection Algorithms Specially Designed for Downlink of CDMA Mobile Radio Systems," *Proc. IEEE Veh. Technol. Conf.*, pp. 203–207.

Kleinrock, L., and Tobagi, F. A. (1975). "Packet Switching in Radio Channels: Part I—Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," *IEEE Trans. Commun.*, vol. COM-23, pp. 1400–1416, December.

Klovsky, D., and Nikolaev, B. (1978) *Sequential Transmission of Digital Information in the Presence of Intersymbol Interference*, Mir Publishers, Moscow.

Kobayashi, H. (1971). "Simultaneous Adaptive Estimation and Decision Algorithm for Carrier Modulated Data Transmission Systems," *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 268–280, June.

Kolmogorov, A. N. (1939). "Sur l'interpolation et extrapolation des suites stationaires," *Comptes Rendus de l'Académie des Sciences*, vol. 208, p. 2043.

Kotelnikov, V. A. (1947). "The Theory of Optimum Noise Immunity," Ph.D. Dissertation, Molotov Energy Institute, Moscow. [Translated by R. A. Silverman, McGraw-Hill, New York.]

Kretzmer, E. R. (1966). "Generalization of a Technique for Binary Data Communication," *IEEE Trans. Commun. Tech.*, vol. COM-14, pp. 67–68, February.

Larsen, K. J. (1973). "Short Convolutional Codes with Maximal Free Distance for Rates 1/2, 1/3, and 1/4," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 371–372, May.

Laurent, P. A. (1986). "Exact and Approximate Construction of Digital Phase Modulations by Superposition of Amplitude Modulated Pulses," *IEEE Trans. Commun.*, vol. COM-34, pp. 150–160, February.

Lee, P. J. (1988). "Construction of Rate $(n-1)/n$ Punctured Convolutional Codes with Minimum Required *SNR* Criterion," *IEEE Trans. Commun.*, vol. 36, pp. 1171–1174, October.

Lee, W. U., and Hill, F. S. (1977). "A Maximum-Likelihood Sequence Estimator with Decision-Feedback Equalizer," *IEEE Trans. Commun.*, vol. 25, pp. 971–979, September.

LeGoff, S., Glavieux, A., and Berrou, C. (1994). "Turbo-codes and High Spectral Efficiency Modulation," *Proc. Int. Conf. Commun.* (*ICC '94*), pp. 645–649, May, New Orleans, LA.

Lender, A. (1963). "The Duobinary Technique for High Speed Data Transmission," *AIEE Trans. Commun. Electronics*, vol. 82, pp. 214–218.

Leon-Garcia, A. (1994). *Probability and Random Processes for Electrical Engineering*, Addison-Wesley, Reading, MA.

Levinson, N. (1947). "The Wiener RMS (Root Mean Square) Error Criterion in Filter Design and Prediction," *J. Math. and Phys.*, vol. 25, pp. 261–278.

Li, J., and Kavehrad, M. (1999). "Effects of Time Selective Multipath Fading on OFDM Systems for Broadband Mobile Applications," *IEEE Commun. Lett.*, vol. 3, pp. 332–334, December.

Li, X., and Cimini, L. (1997). "Effects of Clipping and Filtering on the Performance of OFDM," *Proc. IEEE Vel. Technol. Conf.* (VTC '97), pp. 1634–1638, Phoenix, AZ, May.

Li, X., and Ritcey, J. (1999). "Bit-Interleaved Coded Modulation with Iterative Decoding," in *Commun. 1999, ICC '99 1999 IEEE Int. Conf.*, vol. 2, pp. 858–863.

Li, X., and Ritcey, J. A. (1997). "Bit-Interleaved Coded Modulation with Iterative Decoding," *IEEE Commun. Lett.*, vol. 1, pp. 169–171.

Li, X., and Ritcey, J. A. (1998). "Bit-Interleaved Coded Modulation with Iterative Decoding Using Soft Feedback," *Electronics Lett.*, vol. 34, pp. 942–943.

Li, Y., and Cimini, L. J. (2001). "Bounds on the Interchannel Interference of OFDM in Time-Varying Impairments," *IEEE Trans. Commun.*, vol. 49, pp. 401–404, March.

Lin, S., and Costello, D. J. J. (2004). *Error Control Coding*, 2d ed., Prentice-Hall, Upper Saddle River, NJ.

Linde, Y., Buzo, A., and Gray, R. M. (1980). "An Algorithm for Vector Quantizer Design." *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, January.

Lindell, G. (1985). "On Coded Continuous Phase Modulation," Ph.D. Dissertation, Telecommunication Theory, University of Lund, Lund, Sweden, May.

Lindholm, J. H. (1968). "An Analysis of the Pseudo-Randomness Properties of Subsequences of Long *m*-Sequencies," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 569–576, July.

Lindsey, W. C. (1964). "Error Probabilities for Ricean Fading Multichannel Reception of Binary and *N*-Ary Signals," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 339–350, October.

Lindsey, W. C. (1972). *Synchronization Systems in Communications*, Prentice-Hall, Englewood Cliffs, NJ.

Lindsey, W. C., and Chie, C. M. (1981). "A Survey of Digital Phase-Locked Loops," *Proc. IEEE*, vol. 69, pp. 410–432.

Lindsey, W. C., and Simon, M. K. (1973). *Telecommunication Systems Engineering*, Prentice-Hall, Englewood Cliffs, NJ.

Ling, F., (1988). "Convergence Characteristics of LMS and LS Adaptive Algorithms for Signals with Rank-Deficient Correlation Matrices," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, New York, 25.D.4.7, April.

Ling, F., (1989). "On Training Fractionally-Spaced Equalizers Using Intersymbol Interpolation," *IEEE Trans. Commun.*, vol. 37, pp. 1096–1099, October.

Ling, F., Manolakis, D. G., and Proakis, J. G. (1986a). "Finite, Word-Length Effects in Recursive Least Squares Algorithms with Application to Adaptive Equalization," *Annales des Telecommunications*, vol. 41, pp. 1–9, May/June.

Ling, F., Manolakis, D. G., and Proakis, J. G. (1986b). "Numerically Robust Least-Squares Lattice-Ladder Algorithms with Direct Updating of the Reflection Coefficients," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 837–845, August.

Ling, F., and Proakis, J. G. (1982). Generalized Least Squares Lattice and Its Applications to DFE," *Proc. 1982, IEEE Int. Conf. on Acoust. Speech, Signal Processing*, Paris, France, May.

Ling, F., and Proakis, J. G. (1984a), "Numerical Accuracy and Stability: Two Problems of Adaptive Estimation Algorithms Caused by Round-Off Error," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp. 30.3.1–30.3.4, San Diego, CA. March.

Ling, F., and Proakis, J. G. (1984b). "Nonstationary Learning Characteristics of Least Squares Adaptive Estimation Algorithms," *Proc. Int. Conf. Acoust, Speech, Signal Processing*, pp. 3.7.1–3.7.4, San Diego, CA, March.

Ling, F., and Proakis, J. G. (1984c). "A Generalized Multichannel Least-Squares Lattice Algorithm with Sequential Processing Stages," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 381–389, April.

Ling, F., and Proakis, J. G. (1985). "Adaptive Lattice Decision-Feedback Equalizers—Their Performance and Application to Time-Variant Multipath Channels," *IEEE Trans. Commun*, vol. COM-33, pp. 348–356, April.

Ling, F., and Proakis, J. G. (1986). "A Recursive Modified Gram–Schmidt Algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 829–836, August.

Ling, F., and Qureshi, S. U. H. (1986). "Lattice Predictive Decision-Feedback Equalizer for Digital Communication Over Fading Multipath Channels," *Proc. GLOBECOM '86*, Houston, TX, December.

Ling, F., and Qureshi, S. U. H. (1990). "Convergence and Steady State Behavior of a Phase-Splitting Fractionally Spaced Equalizer," *IEEE Trans. Commun.* vol. 38, pp. 418–425, April.

Ljung, S., and Ljung, L. (1985). "Error Propagation Properties of Recursive Least-Squares Adaptation Algorithms," *Automatica*, vol. 21, pp. 159–167.

Lloyd, S. P. (1982). "Least Squares Quantization in PCM," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 129–137, March.

Loeliger, H. A. (2004). "An Introduction to Factor Graphs," *IEEE Signal Processing Mag.*, vol. 21, pp. 28–41.

Loève, M. (1955). *Probability Theory*, Van Nostrand, Princeton, NJ.

Long, G., Ling, F., and Proakis, J. G. (1987). "Adaptive Transversal Filters with Delayed Coefficient Adaptation," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Dallas, TX, March.

Long, G., Ling, F., and Proakis, J. G. (1988a). "Fractionally-Spaced Equalizers Based on Singular-Value Decomposition," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, New York, 25.D.4.10, April.

Long, G., Ling, F., and Proakis, J. G. (1988b). "Applications of Fractionally-Spaced Decision-Feedback Equalizers to HF Fading Channels," *Proc. MILCOM*, San Diego, CA, October.

Long, G., Ling, F., and Proakis, J. G. (1989). "The LMS Algorithm with Delayed Coefficient Adaptation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, October.

Lu, J., Letaief, K. B., Chuang, J. C., and Liou, M. L. (1999). "M-PSK and M-QAM BER Computation Using Signal-Space Concepts," *IEEE Trans. Commun.*, vol. 47, pp. 181–184, February.

Lucky, R. W. (1965). "Automatic Equalization for Digital Communications, *Bell Syst. Tech. J.,* vol. 44, pp. 547–588, April.

Lucky, R. W. (1966). "Techniques for Adaptive Equalization of Digital Communication," *Bell Syst. Tech. J.*, vol. 45, pp. 255–286.

Lucky, R. W., and Hancock, J. C. (1962). "On the Optimum Performance of N-ary Systems Having Two Degrees of Freedom," *IRE Trans. Commun. Syst.*, vol. CS-10, pp. 185–192, June.

Lucky, R. W., Salz, J., and Weldon, E. J., Jr. (1968). *Principles of Data Communication*, McGraw-Hill, New York.

Lugannani, R. (1969). "Intersymbol Interference and Probability of Error in Digital Systems," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 682–688, November.

Lundgren, C. W., and Rummler, W. D. (1979). "Digital Radio Outage Due to Selective Fading—Observation vs. Prediction from Laboratory Simulation," *Bell Syst. Tech. J.*, vol. 58, pp. 1074–1100, May/June.

Lupas, R., and Verdu, S. (1989). "Linear Multiuser Detectors for Synchronous Code-Division Multiple-Access Channels," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 123–136, January.

Lupas, R., and Verdu, S. (1990). "Near-Far Resistance of Multiuser Detectors in Asynchronous Channels," *IEEE Trans. Commun.*, vol. COM-38, pp. 496–508, April.

MacKay, D. (1999). "Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE Trans. Inform. Theory*, vol. 45, pp. 399–431.

MacKay, D. J. C., and Neal, R. M. (1996). "Near Shannon Limit Performance of Low Density Parity Check Codes," *Electronics Lett.*, vol. 32, pp. 1645–1646.

MacKenchnie, L. R. (1973). "Maximum Likelihood Receivers for Channels Having Memory," Ph.D. Dissertation, Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, January.

MacWilliams, F. J., and Sloane, J. J. (1977). *The Theory of Error Correcting Codes*, North Holland, New York.

Madhow, U., (1998). "Blind Adaptive Interference Suppression for Direct Sequence CDMA," *Proc. IEEE*, vol. 86, pp. 2049–2069, October.

Madhow, U., and Honig, M. L. (1994). "MMSE Interference Suppression for Direct-Sequence Spread-Spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, December.

Magee, F. R., and Proakis, J. G. (1973). "Adaptive Maximum-Likelihood Sequence Estimation for Digital Signaling in the Presence of Intersymbol Interference," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 120–124, January.

Martin, D. R., and McAdam, P. L. (1980). "Convolutional Code Performance with Optimal Jamming," *Conf. Rec. Int. Conf. Commun.*, pp. 4.3.1–4.3.7, May.

Martinez, A., Guillen I Fabregas, A., and Caire, G. (2006). "Error Probability Analysis of Bit-Interleaved Coded Modulation," *IEEE Trans. Inform. Theory*, vol. 52, pp. 262–271.

Massey, J. (1969). "Shift-Register Synthesis and BCH Decoding," *IEEE Trans. Inform. Theory*, vol. 15, pp. 122–127.

Massey, J. L. (1963). *Threshold Decoding*, MIT Press Cambridge, MA.

Massey, J. L. (1965). "Step-by-Step Decoding of the BCH Codes," *IEEE Trans. Inform. Theory*, vol. IT-11, pp. 580–585, October.

Massey, J. L. (1988). "Some New Approaches to Random Access Communications," *Performance '87*, pp. 551–569. [Reprinted 1993 in *Multiple Access Communications*, N. Abramson (ed.), IEEE Press, New York.]

Massey, J. L., and Sain, M. (1968). "Inverses of Linear Sequential Circuits," *IEEE Trans. Comput.*, vol. C-17, pp. 330–337, April.

Matis, K. R., and Modestino, J. W. (1982). "Reduced-State Soft-Decision Trellis Decoding of Linear Block Codes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 61–68, January.

Mazo, J. E. (1975). "Faster-Than-Nyquist Signaling," *Bell Syst. Tech. J.*, vol. 54, pp. 1451–1462, October.

Mazo, J. E. (1979). "On the Independence Theory of Equalizer Convergence," *Bell Syst. Tech. J.*, vol. 58, pp. 963–993, May.

McEliece, R., Rodemich, E., Rumsey, H., and Welch, L. (1977). "New Upper Bounds on the Rate of a Code via the Delsarte-MacWilliams Inequalities," *IEEE Trans. Inform. Theory*, vol. 23, pp. 157–166.

McMahon, M. A. (1984). *The Making of a Profession—A Century of Electrical Engineering in America*, IEEE Press, New York.

Meggitt, J. (1961). "Error Correcting Codes and Their Implementation for Data Transmission Systems," *IEEE Trans. Inform. Theory*, vol. 7, pp. 234–244.

Mengali, U. (1977). "Joint Phase and Timing Acquisition in Data Transmission," *IEEE Trans. Commun.*, vol. COM-25, pp. 1174–1185, October.

Mengali, U., and D'Andrea, A. N. (1997). *Synchronization Techniques for Digital Receivers*, Plenum Press, New York.

Mengali, U., and Morelli, M. (1995). "Decomposition of M-ary CPM Signals into PAM Waveforms," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1265–1275, September.

Meyers, M. H., and Franks, L. E. (1980). "Joint Carrier Phase and Symbol Timing for PAM Systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 1121–1129, August.

Meyr, H., and Ascheid, G. (1990). *Synchronization in Digital Communications*, Wiley Interscience, New York.

Meyr, H., Moenclaey, M., and Fechtel, S. A. (1998). *Digital Commun. Receivers*, Wiley, New York.

Miller, K. S. (1964). *Multidimensional Gaussian Distributions*, Wiley, New York.

Miller, S. L. (1996). "Training Analysis of Adaptive Interference Suppression for Direct-Sequence CDMA Systems," *IEEE Trans. Commun.*, vol. 44, pp. 488–495, April.

Miller, S. L. (1995). "An Adaptive Direct-Sequence Code-Division Multiple Access Receiver for Multiuser Interference Rejection," *IEEE Trans. Commun.*, vol. 43, pp. 1746–1755, Feb./March/April.

Millman, S. (ed.) (1984). *A History of Engineering and Science in the Bell System—Communication Sciences (1925–1980)*, AT&T Bell Laboratories.

Milstein, L. B. (1988). "Interference Rejection in Spread Spectrum Communications," *Proc. IEEE*, vol. 76, pp. 657–671, June.

Mitra, U., and Poor, H. V. (1995). "Adaptive Receiver Algorithm for Near-Far Resistant CDMA," *IEEE Trans. Commun.*, vol. 43, pp. 1713–1724, April.

Miyagaki, Y., Morinaga, N., and Namekawa, T. (1978). "Error Probability Characteristics for CPSK Signal Through m-Distributed Fading Channel," *IEEE Trans. Commun.*, vol. COM-26, pp. 88–100, January.

Moher, M. (1998). "An Iterative Multiuser Decoder for Near-Capacity Communications," *IEEE Trans. Commun.*, vol. 46, pp. 870–880, July.

Moon, J., and Carley, L. R. (1988). "Partial Response Signaling in a Magnetic Recording Channel," vol. MAG-24, pp. 2973–2975, November.

Monsen, P. (1971). "Feedback Equalization for Fading Dispersive Channels," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 56–64, January.

Morf, M. (1977). "Ladder Forms in Estimation and System Identification," *Proc. 11th Annual Asilomar Conf. on Circuits, Systems and Computers*, Monterey, CA, Nov. 7–9.

Morf, M., Dickinson, B., Kailath, T., and Vieira, A. (1977a). "Efficient Solution of Covariance Equations for Linear Prediction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 429–433, October.

Morf, M., and Lee, D. (1978). "Recursive Least Squares Ladder Forms for Fast Parameter Tracking," *Proc. 1978 IEEE Conf. on Decision and Control*, San Diego, CA, pp. 1362–1367, January 12.

Morf, M., Lee, D., Nickolls, J., and Vieira, A. (1977b). "A Classification of Algorithms for ARMA Models and Ladder Realizations," *Proc. 1977 IEEE Int. Conf on Acoustics, Speech, Signal Processing*, Hartford, CT, pp. 13–19, May.

Morf, M., Vieira, A., and Lee, D. (1977c). "Ladder Forms for Identification and Speech Processing," *Proc. 1977 IEEE Conf. on Decision and Control*, New Orleans, LA, pp. 1074–1078, December.

Mueller, K. H., and Muller, M. S. (1976). "Timing Recovery in Digital Synchronous Data Receivers," *IEEE Trans. Commun*, vol. COM-24, pp. 516–531, May.

Mueller, K. H., and Spaulding, D. A. (1975). "Cyclic Equalization—A New Rapidly Converging Equalization Technique for Synchronous Data Communications," *Bell Sys. Tech. J.*, vol. 54, pp. 369–406, February.

Mueller, K. H., and Werner, J. J. (1982). "A Hardware Efficient Passband Equalizer Structure for Data Transmission," *IEEE Trans. Commun.*, vol. COM-30, pp. 438–541, March.

Muller, D. E. (1954). "Application of Boolean Algebra to Switching Circuit Design and to Error Detection," *IRE Trans. Comput.*, vol. EC-3, pp. 6–12, September.

Müller, S., Bäuml, R., Fischer, R., and Huber, J. (1997). "OFDM with Reduced Peak-to-Average Power Ratio by Multiple Signal Representation," *Ann. Telecommun.*, vol. 52, pp. 58–67, February.

Mulligan, M. G. (1988). "Multi-Amplitude Continuous Phase Modulation with Convolutional Coding," Ph.D. Dissertation, Department of Electrical and Computer Engineering, Northeastern University, June.

Nakagami, M. (1960). "The m-Distribution—A General Formula of Intensity Distribution of Rapid Fading," in *Statistical Methods of Radio Wave Propagation*, W. C. Hoffman (ed.), pp. 3–36, Pergamon Press, New York.

Natali, F. D., and Walbesser, W. J. (1969). "Phase-Locked Loop Detection of Binary PSK Signals Utilizing Decision Feedback," *IEEE Trans. Aerospace Electronic Syst.*, vol. AES-5, pp. 83–90, January.

Neeser, F., and Massey, J. (1993). "Proper Complex Random Processes with Applications to Information Theory," *IEEE Trans. Inform. Theory*, vol. 39 pp. 1293–1302, July.

Neyman, J., and Pearson, E. S. (1933). "On the Problem of the Most Efficient Tests of Statistical Hypotheses," *Phil. Trans. Roy. Soc. London, Series A*, vol. 231, pp. 289–337.

Nichols, H., Giordano, A., and Proakis, J. G. (1977). "MLD and MSE Algorithms for Adaptive Detection of Digital Signals in the Presence of Interchannel Interference," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 563–575, September.

North, D. O. (1943). "An Analysis of the Factors Which Determine Signal/Noise Discrimination in Pulse-Carrier Systems," RCA Tech. Report No. 6 PTR-6C.

Nyquist, H. (1924). "Certain Factors Affecting Telegraph Speed," *Bell Syst. Tech. J.*, vol. 3, p. 324.

Nyquist, H. (1928). "Certain Topics in Telegraph Transmission Theory," *AIEE Trans.*, vol. 47, pp. 617–644.

Odenwalder, J. P. (1970). "Optimal Decoding of Convolutional Codes," Ph.D. Dissertation, Department of Systems Sciences, School of Engineering and Applied Sciences, University of California, Los Angeles.

Odenwalder, J. P. (1976). "Dual-*k* Convolutional Codes for Noncoherently Demodulated Channels," *Proc. Int. Telemetering Conf.*, vol. 12, pp. 165–174, September.

Olsen, J. D. (1977). "Nonlinear Binary Sequences with Asymptotically Optimum Periodic Cross Correlation," Ph.D. Dissertation, University of Southern California, December.

Omura, J. (1971). "Optimal Receiver Design for Convolutional Codes and Channels with Memory Via Control Theoretical Concepts," *Inform. Sci.*, vol, 3, pp. 243–266.

Omura, J. K., and Levitt, B. K. (1982). "Code Error Probability Evaluation for Antijam Communication Systems," *IEEE Trans. Commun.*, vol. COM-30, pp. 896–903, May.

Ormeci, P., Liu, X., Goeckel, D., and Wesel, R. (2001). "Adaptive Bit-Interleaved Coded Modulation," *IEEE Trans. Commun.*, vol. 49, pp. 1572–1581.

Osborne, W. P., and Luntz, M. B. (1974). "Coherent and Noncoherent Detection of CPSK," *IEEE Trans. Commun.*, vol. COM-22, pp. 1023–1036, August.

Ozarow, L., Shamai, S., and Wyner, A. (1994). "Information Theoretic Considerations for Cellular Mobile Radio," *IEEE Trans. Veh. Technol.*, vol. 43, pp. 359–378.

Paaske, E. (1974). "Short Binary Convolutional Codes with Maximal Free Distance for Rates 2/3 and 3/4," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 683–689, September.

Paez, M. D., and Glisson, T. H. (1972). "Minimum Mean Squared Error Quantization in Speech PCM and DPCM Systems," *IEEE Trans. Commun.*, vol. COM-20, pp. 225–230, April.

Pahlavan, K. (1985). "Wireless Communications for Office Information Networks," *IEEE Commun. Mag.*, vol. 23, pp. 18–27, June.

Palenius, T. (1991). "On Reduced Complexity Noncoherent Detectors for Continuous Phase Modulation," Ph.D. Dissertation, Telecommunication Theory, University of Lund, Lund, Sweden.

Palenius, T., and Svensson, A. (1993). "Reduced Complexity Detectors for Continuous Phase Modulation Based on Signal Space Approach," *European Trans. Telecommun.*, vol. 4, pp. 51–63, May/June.

Papoulis, A. (1984). *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York.

Papoulis, A., and Pillai, S. (2002). *Probability, Random Variables, and Stochastic Processes*, 4th ed., McGraw-Hill, New York.

Patel, P., and Holtzman, J. (1994). "Analysis of a Simple Successive Interference Cancellation Scheme in a DS/CDMA System," *IEEE J. Select. Areas Commun.*, vol. 12, pp. 796–807, 1994.

Paul, D. B. (1983). "An 800 bps Adaptive Vector Quantization Vocoder Using a Perceptual Distance Measure," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Boston, MA, pp. 73–76, April.

Pearson, K. (1965). *Tables of the Incomplete* Γ*-Function*, Cambridge University Press, London.

Peebles, P. Z. (1987). *Probability, Random Variables, and Random Signal Principles*, McGraw-Hill, New York.

Peel, C. B., Hochwald, B. M., and Swindlehurst, A. L. (2005). "A Vector-Perturbation Technique for Near Capacity Multiantenna Multiuser Communication—Part I: Channel Inversion and Regularization," *IEEE Trans. Commun.*, vol. 53, pp. 195–202, January.

Peterson, K., and Tarokh, V. (2000). "On the Existence and Construction of Good Codes with Low Peak-to-Average Power Ratios," *IEEE Trans. Inform. Theory*, vol. 46, pp. 1974–1986, September.

Peterson, R. L., Ziemer, R. E., and Borth, D. E. (1995). *Introduction to Spread Spectrum Communications*, Prentice-Hall, Upper Saddle River, NJ.

Peterson, W. W. (1960). "Encoding and Error-Correction Procedures for Bose–Chaudhuri Codes," *IRE Trans. Inform. Theory*, vol. IT-6, pp. 459–470, September.

Peterson, W. W., and Weldon, E. J., Jr. (1972). *Error-Correcting Codes*, 2nd ed., MIT Press, Cambridge, MA.

Picchi, G., and Prati, G. (1987). "Blind Equalization and Carrier Recovery Using a Stop-and-Go Decision Directed Algorithm," *IEEE Trans. Commun.*, vol. COM-35, pp. 877–887, September.

Picinbono, B. (1978). "Adaptive Signal Processing for Detection and Communication," in *Communication Systems and Random Process Theory*, J. K. Skwirzynski (ed.), Sijthoff & Nordhoff, Alphen aan den Rijn, The Netherlands.

Pickholtz, R. L., Schilling, D. L., and Milstein, L. B. (1982). "Theory of Spread Spectrum Communications—A Tutorial," *IEEE Trans. Commun.*, vol. COM-30, pp. 855–884, May.

Pieper, J. F., Proakis, J. G., Reed, R. R., and Wolf, J. K. (1978). "Design of Efficient Coding and Modulation for a Rayleigh Fading Channel," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 457–468, July.

Pierce, J. N. (1958). "Theoretical Diversity Improvement in Frequency-Shift Keying," *Proc. IRE*, vol. 46, pp. 903–910, May.

Pierce, J. N., and Stein, S. (1960). " Multiple Diversity with Non-Independent Fading," *Proc. IRE*, vol. 48, pp. 89–104, January.

Plotkin, M. (1960). "Binary Codes with Specified Minimum Distance," *IRE Trans. Inform. Theory*, vol. IT-6, pp. 445–450, September.

Poor, H. V., and Rusch, L. A. (1994). "Narrowband Interference Suppression in Spread Spectrum CDMA," *IEEE Personal Commun.*, vol. 1, pp. 14–27, Third Quarter.

Poor, H. V., and Verdu, S. (1988). "Single-User Detectors for Multiuser Channels," *IEEE Trans. Commun.*, vol. 36, pp. 50–60, January.

Popovic, B. M. (1991). "Synthesis of Power Efficient Multitone Signals with Flat Amplitude Spectrum," *IEEE Trans. Commun.*, vol. 39, pp. 1031–1033, July.

Prange, E. (1957). "Cyclic Error Correcting Codes in Two Symbols," Tech. Rep. TN-57-103, Air Force Cambridge Research Center, Cambridge, MA.

Price, R. (1954). "The Detection of Signals Perturbed by Scatter and Noise," *IRE Trans. Inform. Theory*, vol. PGIT-4, pp. 163–170, September.

Price, R. (1956). "Optimum Detection of Random Signals in Noise, with Application to Scatter-Multipath Communication," *IRE Trans. Inform. Theory*, vol. IT-2, pp. 125–135, December.

Price, R. (1962a). "Error Probabilities for Adaptive Multichannel Reception of Binary Signals," MIT Lincoln Laboratory, Lexington, MA, Techn. Report No. 258, July.

Price, R. (1962b). "Error Probabilities for Adaptive Multichannel Reception of Binary Signals," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 305–316, September.

Price, R. (1972). "Nonlinearly Feedback-Equalized PAM vs. Capacity," *Proc. 1972 IEEE Int. Conf. on Commun.* Philadelphia, PA, pp. 22.12–22.17, June.

Price, R., and Green, P. E., Jr. (1958). "A Communication Technique for Multipath Channels," *Proc. IRE*, vol. 46, pp. 555–570, March.

Price, R., and Green, P. E., Jr. (1960). "Signal Processing in Radar Astronomy—Communication via Fluctuating Multipath Media," MIT Lincoln Laboratory, Lexington, MA, Tech. Report No. 234, October.

Proakis, J. G. (1968). "Probabilities of Error for Adaptive Reception of M-Phase Signals," *IEEE Trans. Commun. Tech.*, vol. COM-16, pp. 71–81, February.

Proakis, J. G. (1970). "Adaptive Digital Filters for Equalization of Telephone Channels," *IEEE Trans. Audio and Electroacoustics*, vol. AU-18, pp. 195–200, June.

Proakis, J. G. (1975). "Advances in Equalization for Intersymbol Interference," in *Advances in Communication Systems*, vol. 4, A. J. Viterbi (ed.), Academic, New York.

Proakis, J. G. (1998). "Equalization Techniques for High-Density Magnetic Recording," *IEEE Signal Processing Mag.*, vol. 15, pp. 73–82, July.

Proakis, J. G., Drouilhet, P. R., Jr., and Price, R. (1964). "Performance of Coherent Detection Systems Using Decision-Directed Channel Measurement," *IEEE Trans. Commun. Syst.*, vol. CS-12, pp. 54–63, March.

Proakis, J. G., and Ling, F. (1984). "'Recursive Least Squares Algorithms for Adaptive Equalization of Time-Variant Multipath Channels," *Proc. Int. Conf. Commun.* Amsterdam, The Netherlands, May.

Proakis, J. G., and Manolakis, D. G. (2006). *Introduction to Digital Processing*, Prentice-Hall, Upper Saddle River, NJ, 2nd Ed.

Proakis, J. G., and Miller, J. H. (1969). "Adaptive Receiver for Digital Signaling through Channels with Intersymbol Interference," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 484–497, July.

Proakis, J. G., and Rahman, I. (1979). "Performance of Concatenated Dual-k Codes on a Rayleigh Fading Channel with a Bandwidth Constraint," *IEEE Trans. Commun.*, vol. COM-27, pp. 801–806, May.

Pursley, M. B. (1979). "On the Mean-Square Partial Correlation of Periodic Sequences," *Proc. 1979 Conf. Inform. Science and Systems*, Johns Hopkins University, Baltimore, MD., pp. 377–379, March.

Qureshi, S. U. H. (1976). "Timing Recovery for Equalized Partial Response Systems," *IEEE Trans. Commun.*, vol. COM-24, pp. 1326–1331, December.

Qureshi, S. U. H. (1977). "Fast Start-up Equalization with Periodic Training Sequences," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 553–563, September.

Qureshi, S. U. H. (1985). "Adaptive Equalization," *Proc. IEEE*, vol. 53, pp. 1349–1387, September.

Qureshi, S. U. H., and Forney, G. D., Jr. (1977). "Performance and Properties of a $T/2$ Equalizer," *Natl. Telecom. Conf. Record*, pp. 11.1.1–11.1.14, Los Angeles, CA. December.

Rabiner, L. R., and Schafer, R. W. (1978). *Digital Processing of Speech Signals*, Prentice-Hall, Englewood Cliffs, NJ.

Radon, J. (1922) "Lineare Scharen Orthogonaler Matrizen," *Abhandlungen aus dem Mathimatischen Seminar der Hamburgishen Universitat*, pp. 1–14.

Raheli, R., Polydoros, A., and Tzou, C. K. (1995). "Per-Survivor Processing: A General Approach to MLSE in Uncertain Environment," *IEEE Trans. Commun.*, vol. 43, pp. 354–364, Feb./March/April.

Rahman, I. (1981). "Bandwidth Constrained Signal Design for Digital Communication over Rayleigh Fading Channels and Partial Band Interference Channels," Ph.D. Dissertation, Department of Electrical Engineering, Northeastern University, Boston, MA.

Ramsey, J. L. (1970). "Realization of Optimum Interleavers," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 338–345.

Rapajic, P. B., and Vucetic, B. S. (1994). "Adaptive Receiver Structures for Asynchronous CDMA Systems," *IEEE J. Select. Areas Commun*, vol. 12, pp. 685–697, May.

Raphaeli, D., and Zarai, Y. (1998). "Combined Turbo Equalization and Turbo Decoding," *IEEE Commun. Letters*, vol. 2, pp. 107–109, April.

Rappaport, T. S. (1996). *Wireless Commun.*, Prentice-Hall, Upper Saddle River, NJ.

Reed, I. S. (1954). "A Class of Multiple-Error Correcting Codes and the Decoding Scheme," *IRE Trans. Inform.*, vol. IT-4, pp. 38–49, September.

Reed, I. S., and Solomon, G. (1960). "Polynomial Codes Over Certain Finite Fields," *SIAM J.*, vol. 8, pp. 300–304, June.

Reed, M. C., Schlegel, C. B., Alexander, P. D., and Asenstorfer, J. A. (1998). "Iterative Multiuser Detection for CDMA with FEC: Near Single User Performance," *IEEE Trans. Commun.*, vol. 46, pp. 1693–1699, December.

Rimoldi, B. E. (1989). "Design of Coded CPFSK Modulation Systems for Bandwidth and Energy Efficiency," *IEEE Trans. Commun.*, vol. 37, pp. 897–905, September.

Rimoldi, B. E. (1988). "A Decomposition Approach to CPM," *IEEE Trans. Inform. Theory*, vol. 34, pp. 260–270, March.

Rizos, A. D., Proakis, J. G., and Nguyen, T. Q. (1994). "Comparison of DFT and Cosine Modulated Filter Banks in Multicarrier Modulation," *Proc. Globecom'94*, pp. 687–691, San Francisco, CA, November.

Roberts, L. G. (1975). "Aloha Packet System with and without Slots and Capture," *Comp. Commun. Rev.*, vol. 5, pp. 28–42, April.

Robertson, P., and Hoeher, P. (1997). "Optimal and Sub-Optimal Maximum a Posteriori Algorithms Suitable for Turbo Decoding," *European Trans. Telecommun.*, vol. 8, pp. 119–125.

Robertson, P., and Kaiser, S. (1999). "Analysis of the Loss of Orthogonality through Doppler Spread in OFDM Systems," *Proc. IEEE Globecom*, pp. 701–706, December.

Robertson, P., Villebrun, E., and Hoeher, P. (1995). "A Comparison of Optimal and Sub-Optimal MAP Decoding Algorithms Operating in the Log Domain," in *Proc. IEEE Int. Conf. Communic. (ICC)*, pp. 1009–1013, IEEE, Seattle, BC, Canada.

Robertson, P., and Wörz, T. (1998). "Bandwidth-Efficient Turbo Trellis-Coded Modulation Using Punctured Component Codes," *IEEE J. Selected Areas, Commun.*, vol. 16, pp. 206–218, February.

Rowe, H. E., and Prabhu, V. K. (1975). "Power Spectrum of a Digital Frequency Modulation Signal," *Bell Syst. Tech. J.*, vol. 54, pp. 1095–1125, July/August.

Rummler, W. D. (1979). "A New Selective Fading Model: Application to Propagation Data," *Bell Syst. Tech. J.*, vol. 58, pp. 1037–1071, May/June.

Rusch, L. A., and Poor, H. V. (1994). "Narrowband Interference Suppression in CDMA Spread Spectrum Communications," *IEEE Trans. Commun.*, vol. 42, pp. 1969–1979, April.

Ryan, W. E. (2003). "Concatenated Convolutional Codes and Iterative Decoding," in *Wiley Encyclopedia of Telecommunications*, J. G. Proakis (ed.), Wiley, New York.

Ryder, J. D., and Fink, D. G. (1984). *Engineers and Electronics*, IEEE Press, New York.

Salehi, M. (1992). "Capacity and Coding for Memories with Real-Time Noisy Defect Information at Encoder and Decoder," *IEEE Proc. Commun., Speech and Vision*, vol. 139, pp. 113–117.

Salehi, M., and Proakis, J. G. (1995). "Coded Modulation Techniques for Cellular Mobile Systems," in *Worldwide Wireless Communications*, F. S. Barnes (ed.), pp. 215–238, International Engineering Consortium, Chicago, IL.

Saltzberg, B. R. (1967). "Performance of an Efficient Parallel Data Transmission System," *IEEE Trans. Commun.*, vol. COM-15, pp. 805–811, December.

Saltzberg, B. R. (1968). "Intersymbol Interference Error Bounds with Application to Ideal Bandlimited Signaling," *IEEE Trans. Inform. Theory*, vol. IT-14, pp. 563–568, July.

Salz, J. (1973). "Optimum Mean-Square Decision Feedback Equalization," *Bell Syst. Tech. J.*, vol. 52, pp. 1341–1373, October.

Salz, J., Sheehan, J. R., and Paris, D. J. (1971). "Data Transmission by Combined AM and PM," *Bell Syst. Tech. J.*, vol. 50, pp. 2399–2419, September.

Sarwate, D. V., and Pursley, M. B. (1980). "Crosscorrelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE*, vol. 68, pp. 593–619, May.

Sason, I., and Shamai, S. (2000). "Improved Upper Bounds on the ML Decoding Error Probability of Parallel and Serial Concatenated Turbo Codes via Their Ensemble Distance Spectrum," *IEEE Trans. Inform. Theory*, vol. 46, pp. 24–47.

Sason, I., and Shamai, S. (2001a). "On Gallager-Type Bounds for the Mismatched Decoding Regime with Applications to Turbo Codes," in *Proc. 2001 IEEE Int. Symp. Inform. Theory*, p. 134.

Sason, I., and Shamai, S. (2001b). "On Improved Bounds on the Decoding Error Probability of Block Codes over Interleaved Fading Channels, with Applications to Turbo-like Codes," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2275–2299.

Sato, Y. (1975). "A Method of Self-Recovering Equalization for Multilevel Amplitude-Modulation Systems," *IEEE Trans. Commun*, vol. COM-23, pp. 679–682, June.

Sato, Y. et al. (1986). "Blind Suppression of Time Dependency and Its Extension to Multi-Dimensional Equalization," *Proc. ICC'86*, pp. 46.4.1–46.4.5.

Sato, Y. (1994). "Blind Equalization and Blind Sequence Estimation," *IEICE Trans. Commun.*, vol. E77-b, pp. 545–556, May.

Satorius, E. H., and Alexander, S. T. (1979). "Channel Equalization Using Adaptive Lattice Algorithms," *IEEE Trans. Commun.*, vol. COM-27, pp. 899–905, June.

Satorius, E. H., and Pack, J. D. (1981). "Application of Least Squares Lattice Algorithms to Adaptive Equalization," *IEEE Trans. Commun.*, vol. COM-29, pp. 136–142, February.

Savage, J. E. (1966). "Sequential Decoding—The Computation Problem," *Bell Syst. Tech. J.*, vol. 45, pp. 149–176, January.

Schlegel, C. (1997). *Trellis Coding*, IEEE Press, New York.

Schlegel, C., and Costello, D. J. J. (1989). "Bandwidth Efficient Coding for Fading Channels: Code Construction and Performance Analysis," *IEEE J. Selected Areas Commun.*, vol. SAC-7, pp. 1356–1368.

Scholtz, R. A. (1977). "The Spread Spectrum Concept," *IEEE Trans. Commun.*, vol. COM-25, pp. 748–755, August.

Scholtz, R. A. (1979). "Optimal CDMA Codes, *1979 Nat. Telecommun. Conf. Rec.*, Washington, DC, pp. 54.2.1–54.2.4, November.

Scholtz, R. A. (1982). "The Origins of Spread Spectrum," *IEEE Trans. Commun.*, vol. COM-30, pp. 822–854, May.

Schonhoff, T. A. (1976). "Symbol Error Probabilities for M-ary CPFSK: Coherent and Noncoherent Detection," *IEEE Trans. Commun.*, vol. COM-24, pp. 644–652, June.

Seshadri, N. (1994). "Joint Data and Channel Estimation Using Fast Blind Trellis Search Techniques," *IEEE Trans. Commun.*, vol. COM-42, pp. 1000–1011, March.

Seshadri, N., and Winters, J. H. (1994). "Two Schemes for Improving the Performance of Frequency Division Duplex (FDD) Transmission Systems Using Transmitter Antenna Diversity," *Intern. J. Wireless Inform. Networks*, vol. 1, pp. 49–60, January.

Shalvi, O., and Weinstein, E. (1990). "New Criteria for Blind Equalization of Nonminimum Phase Systems Channels," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 312–321, March.

Shannon, C. E. (1948a). "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, July.

Shannon, C. E. (1948b). "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, vol. 27, pp. 623–656, October.

Shannon, C. E. (1949). "Communication in the Presence of Noise," *Proc. IRE*, vol. 37, pp. 10–21, January.

Shannon, C. E. (1958). "Channels with Side Information at the Transmitter," *IBM J. Res. and Deve.*, vol. 2, pp. 289–293.

Shannon, C. E. (1959a). "Coding Theorems for a Discrete Source with a Fidelity Criterion," *IRE Nat. Conv. Rec.*, pt. 4, pp. 142–163, March.

Shannon, C. E. (1959b). "Probability of Error for Optimal Codes in a Gaussian Channel," *Bell Syst. Tech. J.*, vol. 38, pp. 611–656, May.

Shannon, C. E., Gallager, R. G., and Berlekamp, E. R. (1967). "Lower Bounds to Error Probability for Coding on Discrete Memoryless Channels, I and II," *Inform. Control.*, vol. 10, pp. 65–103, January; pp. 527–552, May.

Shimbo, O., and Celebiler, M. (1971). "The Probability of Error due to Intersymbol Interference and Gaussian Noise in Digital Communication Systems," *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 113–119, April.

Siegel, P. H., and Wolf, J. K. (1991). "Modulation and Coding for Information Storage," *IEEE Commun. Mag.* vol. 30, pp. 68–86, December.

Simmons, S. J., and Wittke, P. H. (1983). "Low Complexity Decoders for Constant Envelope Digital Modulation," *IEEE Trans. Commun.*, vol. 31, pp. 1273–1280, December.

Simon, M., and Alouini, M. (1998). "A Unified Approach to Performance Analysis of Digital Communication over Generalized Fading Channels," *Proc. IEEE*, vol. 48, pp. 1860–1877, September.

Simon, M. K., and Alouini, M. S. (2000). *Digital Communication over Fading Channels: A Unified Approach to Performance Analysis*, Wiley, New York.

Simon, M. K., and Divsalar, D. (1985). "Combined Trellis Coding with Asymmetric MPSK Modulation," *JPL Publ. 85–24*, Pasadena, CA, May.

Simon, M. K., Hinedi, S., and Lindsey, W. C. (1995). *Digital Commun. Techniques*, Prentice-Hall: Upper Saddle River, NJ.

Simon, M. K., Omura, J. K., Scholtz, R. A., and Levitt, B. K. (1985). *Spread Spectrum Communications Vol. I, II, III,* Computer Science Press, Rockville, MD.

Simon, M. K., Omura, J. K., Scholtz, R. A., and Levitt, B. K. (1994). *Spread Spectrum Communications Handbook*, New York: McGraw-Hill.

Simon, M. K., and Smith, J. G. (1973). "Hexagonal Multiple Phase-and-Amplitude-Shift Keyed Signal Sets," *IEEE Trans. Commun.*, vol. COM-21, pp. 1108–1115, October.

Slepian, D. (1956). "A Class of Binary Signaling Alphabets," *Bell Syst. Tech. J.*, vol. 35, pp. 203–234, January.

Slepian, D. (1974). *Key Papers in the Development of Information Theory*, IEEE Press, New York.

Slepian, D., and Wolf, J. K. (1973). "A Coding Theorem for Multiple Access Channels with Correlated Sources," *Bell Syst. Tech. J.*, vol. 52, pp. 1037–1076.

Sloane, N. J. A., and Wyner, A. D. (1993). *The Collected Papers of Shannon*, IEEE Press, New York.

Slock, D. T. M., and Kailath, T. (1991). "Numerically Stable Fast Transversal Filters for Recursive Least-Squares Adaptive Filtering" *IEEE Trans. Signal Processing*, SP-39, pp. 92–114, January.

Smith, J. W. (1965). "The Joint Optimization of Transmitted Signal and Receiving Filter for Data Transmission Systems," *Bell Syst. Tech. J.*, vol. 44, pp. 1921–1942, December.

Stamoulis, A., Diggavi, S. N., and Al-Dhahir, N. (2002). "Intercarrier Interference in MIMO OFDM," *IEEE Trans. Signal Proc.*, vol. 50, pp. 2451–2464, October.

Stark, H., and Woods, J. W. (2002). *Probability, Random Processes and Estimation Theory for Engineers*, 3rd ed., Prentice-Hall, Upper Saddle River, NJ.

Starr, T., Cioffi, J. M., and Silverman, P. J. (1999). *Digital Subscriber Line Technology*, Prentice-Hall, Upper Saddle River, NJ.

Stenbit, J. P. (1964). Table of Generators for BCH Codes," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 390–391, October.

Stiffler, J. J. (1971). *Theory of Synchronous Communications*, Prentice-Hall, Englewood Cliffs, NJ.

Stuber, G. L. (1996). *Principles of Mobile Communications*, Kluwer Academic Publishers, Boston.

Sundberg, C. E. (1986). "Continuous Phase Modulation," *IEEE Commun. Mag.*, vol. 24, pp. 25–38, April.

Sundberg, C.-E. W., and Seshadri, N. (1993). "Coded Modulation for Fading Channels: An Overview," *European Trans. Telecommun.*, vol. 4, pp. 309–324.

Suzuki, H. (1977). "A Statistical Model for Urban Multipath Channels with Random Delay," *IEEE Trans. Commun*, vol. COM-25, pp. 673–680, July.

Svensson, A. (1984). "Receivers for CPM", Ph.D. Dissertation, Telecommunication Theory, University of Lund, Lund, Sweden.

Svensson, A., and Sundberg C.W. (1983). "Optimized Reduced-Complexity Viterbi Detectors for CPM," *Proc. GLOBECOM'83*, pp. 22.1.1–22.1.8, San Diego, CA.

Svensson, A., Sundberg, C.W., and Aulin, T. (1984). "A Class of Reduced Complexity Viterbi Detectors for Partial Response Continuous Phase Modulation," *IEEE Trans. Commun.*, vol. 32, pp. 1079–1087, October.

Tang, D. L., and Bahl, L. R. (1970). "Block Codes for a Class of Constrained Noiseless Channels," *Inform. Control*, vol. 17, pp. 436–461.

Tanner, R. (1981). "A Recursive Approach to Low Complexity Codes," *IEEE Trans. Inform. Theory*, vol. 27, pp. 533–547.

Tao, M., and Cheng, R. S. (2001). "Differential Space-Time Block Codes," *Proc. IEEE Globecom.*, vol. 2, pp. 1098–1102, November.

Taricco, G., and Elia, M. (1997). "Capacity of Fading Channel with No Side Information," in *Electronics Lett.*, vol. 33, pp. 1368–1370.

Tarokh, V., and Jafarkhani, H., (2000). "A Differential Detection Scheme for Transmit Diversity," *IEEE J. Selected Areas Commun.*, vol. 18, pp. 1169–1174, July.

Tarokh, V., and Jafarkhani, H. (2000). "On the Computation and Reduction of the Peak-to-Average Power Ratio in Multicarrier Communications," *IEEE Trans. Commun.*, vol. 48, pp. 37–44, January.

Tarokh, V., Seshadri, N., and Calderbank, A. R. (1998). "Space-Time Codes for High Data Rate Wireless Communication: Performance Analysis and Code Construction," *IEEE Trans. Inform. Theory*, vol. IT-44, pp. 744–765, March.

Tarokh, V., Jafarkhani, H., and Calderbank, A. R. (1999a). "Space-Time Block Codes from Orthogonal Designs," *IEEE Trans. Inform. Theory*, vol. IT-45, pp. 1456–1467, July.

Tarokh, V., Naguib, A., Seshadri, N., and Calderbank, A. R. (1999b). "Space-Time Codes for High Data Rate Wireless Communication: Performance Criteria in the Presence of Channel Estimation Errors, Mobility and Multiple Paths," *IEEE Trans. Commun.*, vol. COM-47, pp. 199–207, February.

Tarokh, V., Jafarkhani, H., and Calderbank, A. R. (1999c). "Space-Time Block Coding for Wireless Communications: Performance Results," *IEEE J. Selected Areas on Commun.*, vol. JSAC-17, pp. 451–460, March.

Tausworth, R. C., and Welch, L. R. (1961). "Power Spectra of Signals Modulated by Random and Pseudorandom Sequences," *JPL Tech. Rep. 32–140*, October 10.

Taylor, D. P., Vitetta, G. M., Hart, B. D., and Mammala, A. (1998). "Wireless Channel Equalization," *European Trans. Telecommun. (ETT)*, vol. 9, pp. 117–143, March/April.

Telatar, I. E. (1999). "Capacity of Multi-Antenna Gaussian Channels," *European Trans. Telecomm.*, vol. 10, pp. 585–595, November/December.

Tellado, J., and Cioffi, J. M. (1998). "Efficient Algorithms for Reducing PAR in Multicarrier Systems," *Proc. 1998 IEEE Int. Symp. Inform. Theory*, p. 191, August 16–21, Cambridge, MA. Also in *Proc. 1998 GLOBECOM*, Nov. 8–12, Sydney, Australia.

ten Brink, S. (2001). "Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes," *IEEE Trans. Commun.*, vol. 49, pp. 1727–1737.

Tietäväinen, A. (1973). "On the Nonexistence of Perfect Codes over Finite Fields," *SIAM J. Applied Math.*, vol. 24, pp. 88–96.

Thomas, C. M., Weidner, M. Y., and Durrani, S. H. (1974). "Digital Amplitude-Phase-Keying with M-ary Alphabets," *IEEE Trans. Commun.*, vol. COM-22, pp. 168–180, February.

Tomlinson, M. (1971). "A New Automatic Equalizer Employing Modulo Arithmetic," *Electr. Lett.*, vol. 7, pp. 138–139.

Tong, L., Xu, G., Hassibi, B., and Kailath, T. (1995). "Blind Channel Identification Based on Second-Order Statistics: A Frequency-Domain Approach," *IEEE Trans. Inform. Theory*, vol. IT-41, pp. 329–334, January.

Tong, L., Xu, G., and Kailath, T. (1994). "Blind Identification and Equalization Based on Second-Order Statistics," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 340–349, March.

Tse, D., and Viswanath, P. (2005). *Fundamentals of Wireless Communication*, Cambridge University Press, Cambridge, U.K.

Tufts, D. W. (1965). "Nyquist's Problem—The Joint Optimization of Transmitter and Receiver in Pulse Amplitude Modulation," *Proc. IEEE*, vol. 53, pp. 248–259, March.

Tulino, A. M., and Verdu, S. (2004). *Ramdom Matrix Theory and Wireless Communications*, New Publishers, Inc., June 28.

Turin, G. L. (1961). "On Optimal Diversity Reception," *IRE Trans. Inform. Theory*, vol. IT-7, pp. 154–166, July.

Turin, G. L. (1962). "On Optimal Diversity Reception II," *IRE Trans. Commun. Syst.*, vol. CS-12, pp. 22–31, March.

Turin, G. L. et al. (1972). "Simulation of Urban Vehicle Monitoring Systems," *IEEE Trans. Vehicular Tech.*, pp. 9–16, February.

Tyner, D. J., and Proakis, J. G. (1993). "Partial Response Equalizer Performance in Digital Magnetic Recording Channels," *IEEE Trans. Magnetics*, vol. 29, pp. 4194–4208, November.

Tzannes, M. A., Tzannes, M. C., Proakis, J. G., and Heller, P. N. (1994). "DMT Systems, DWMT Systems and Digital Filter Banks," *Proc. Int. Conf. Commun.*, pp. 31–315, New Orleans, LA, May 1–5.

Ungerboeck, G. (1972). "Theory on the Speed of Convergence in Adaptive Equalizers for Digital Communication," *IBM J. Res. Dev.*, vol. 16, pp. 546–555, November.

Ungerboeck, G. (1974). "Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems," *IEEE Trans. Commun.*, vol. COM-22, pp. 624–636, May.

Ungerboeck, G. (1976). "Fractional Tap-Spacing Equalizer and Consequences for Clock Recovery in Data Modems," *IEEE Trans. Commun.*, vol. COM-24, pp. 856–864, August.

Ungerboeck, G. (1982). "Channel Coding with Multilevel/Phase Signals," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55–67, January.

Ungerboeck, G. (1987). "Trellis-Coded Modulation with Redundant Signal Sets, Parts I and II," *IEEE Commun. Mag.*, vol. 25, pp. 5–21, February.

Ungerboeck, G., and Csajka, I. (1976). "On Improving Data-Link Performance by Increasing the Channel Alphabet and Introducing Sequence Coding," *1976 Int. Conf. Inform. Theory, Ronneby, Sweden*, June.

Vaidyanathan, P. P. (1993). *Multirate Systems and Filter Banks*, Prentice-Hall, Englewood Cliffs, NJ.

Van Etten, W. (1975). "An Optimum Linear Receiver for Multiple Channel Digital Transmission Systems," *IEEE Trans. Commun.*, vol. COM-23, pp. 828–834, August.

Van Etten, W. (1976). "Maximum Likelihood Receiver for Multiple Channel Transmission Systems," *IEEE Trans. Commun.*, vol. COM-24, pp. 276–283, February.

Van Trees, H. L. (1968). *Detection, Estimation, and Modulation Theory,* vol. I, Wiley, New York.

Varanasi, M. K. (1999). "Decision Feedback Multiuser Detection: A Systematic Approach," *IEEE Trans. Inform. Theory*, vol. 45, pp. 219–240, January.

Varanasi, M. K., and Aazhang, B. (1990). "Multistage Detection in Asynchronous Code-Division Multiple Access Communications," *IEEE Trans. Commun.*, vol. 38, pp. 509–519, April.

Varsharmov, R. R. (1957). "Estimate of the Number of Signals in Error Correcting Codes," *Doklady Akad. Nauk, S.S.S.R.*, vol. 117, pp. 739–741.

Verdu, S. (1986a). "Minimum Probability of Error for Asynchronous Gaussian Multiple-Access Channels," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 85–96, January.

Verdu, S. (1986b). "Multiple-Access Channels with Point-Process Observation: Optimum Demodulation," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 642–651, September.

Verdu, S. (1986c). "Optimum Multiuser Asymptotic Efficiency," *IEEE Trans. Commun.*, vol. COM-34, pp. 890–897, September.

Verdu, S. (1989). "'Recent Progress in Multiuser Detection," *Advances in Communications and Signal Processing*, Springer-Verlag, Berlin. [Reprinted in *Multiple Access Communications*, N. Abramson (ed.), IEEE Press, New York.]

Verdu, S. (1998). *Multiuser Detection*, Cambridge University Press, New York.

Verdu, S. (1998). "Fifty Years of Information Theory," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2057–2078, October.

Verdu, S., and Han, T., (1994). "A General Formula for Channel Capacity," IEEE Transactions on Information Theory, vol. IT-40, No. 4, pp. 1147–1157, July.

Verhoeff, T. (1987). "An Updated Table of Minimum-Distance Bounds for Binary Linear Codes," *IEEE Trans. Inform. Theory*, vol. 33, pp. 665–680.

Vermeulen, F. L., and Hellman, M. E. (1974). "Reduced-State Viterbi Decoders for Channels with Intersymbol Interference," *Conf. Rec. ICC '74*, pp. 37B.1–37B.4, June, Minneapolis, MN.

Vijayan, R., and Poor, H. V. (1990). "Nonlinear Techniques for Interference Suppression in Spread Spectrum Systems," *IEEE Trans. Commun*, vol. 38, pp. 1060–1065, July.

Vishwanath, S., Jindal, N., and Goldsmith, A. (2003). "Duality, Achievable Rates, and Sum Capacity of Gaussian MIMO Broadcast Channels," *IEEE Trans. Inform. Theory,* vol. 49, pp. 2658–2668, August.

Viswanath, P., and Tse, D. (2003). "Sum Capacity of the Vector Gaussian Broadcast Channel and Uplink-Downlink Duality," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1912–1921, August.

Viterbi, A. J. (1966). *Principles of Coherent Communication*, McGraw-Hill, New York.

Viterbi, A. J. (1967). "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 260–269, April.

Viterbi, A. J. (1969). “Error Bounds for White Gaussian and Other Very Noisy Memoryless Channels with Generalized Decision Regions,” *IEEE Trans. Inform. Theory*, vol., IT-15, pp. 279–287, March.

Viterbi, A. J. (1971). “Convolutional Codes and Their Performance in Communication Systems,” *IEEE Trans. Commun. Tech.*, vol. COM-19, pp. 751–772, October.

Viterbi, A. J. (1978). “A Processing Satellite Transponder for Multiple Access by Low-Rate Mobile Users,” *Proc. Fourth Int. Conf. on Digital Satellite Communications*, Montreal, Canada, pp. 166–174, October.

Viterbi, A. J. (1979). “Spread Spectrum Communication—Myths and Realities,” *IEEE Commun. Mag.*, vol. 17, pp. 11–18, May.

Viterbi, A. J. (1985). “When Not to Spread Spectrum—A Sequel,” *IEEE Commun. Mag.*, vol. 23, pp. 12–17, April.

Viterbi, A. J. (1995). *CDMA: Principles of Spread Spectrum Communications*, Addison-Wesley, Reading, MA.

Viterbi, A. J. (1990). “Very Low Rate Convolutional Codes for Maximum Theoretical Performance of Spread-Spectrum Multiple-Access Channels,” *IEEE J. Selected Areas Commun.*, vol. 8, pp. 641–649, May.

Viterbi, A. J., and Jacobs, I. M. (1975). “Advances in Coding and Modulation for Noncoherent Channels Affected by Fading, Partial Band, and Multiple-Access Interference,” in *Advances in Communication Systems*, vol. 4, A. J. Viterbi (ed.), Academic, New York.

Viterbi, A. J., and Omura, J. K. (1979). *Principles of Digital Communication and Coding*, McGraw-Hill, New York.

Viterbi, A. J., Wolf, J. K., Zehavi, E., and Padovani, R. (1989). “A Pragmatic Approach to Trellis-Coded Modulation,” *IEEE Commun. Mag.*, vol. 27, pp. 11–19, July.

Viterbo, E., and Boutros, J. (1999). “A Universal Lattice Code Decoder for Fading Channels,” *IEEE Trans. Inform. Theory*, vol. 45, pp. 1639–1642, July.

Wainberg, S., and Wolf, J. K. (1970). “Subsequences of Pseudo-Random Sequences,” *IEEE Trans. Commun. Tech.*, vol. COM-18, pp. 606–612, October.

Wainberg, S., and Wolf, J. K. (1973). “Algebraic Decoding of Block Codes Over a q-ary Input, Q-ary Output Channel, $Q > q$,” *Inform. Control*, vol. 22, pp. 232–247, April.

Wald, A. (1947). *Sequential Analysis*, Wiley, New York.

Wang, H., and Xia, X. G. (2003). “Upper Bounds of Rates of Space-Time Block Codes from Complex Orthogonal Designs,” *IEEE Trans. Inform. Theory*, vol. 49, pp. 2788–2796, October.

Wang, T., Proakis, J. G., Masry, E., and Zeidler, J. R. (2006). “Performance Degradation of OFDM Systems due to Doppler Spreading,” *IEEE Trans. Wireless Commun.*, vol. 5, pp. 1422–1432, June.

Wang, X., and Poor, H. V. (1998a). “Blind Equalization and Multiuser Detection for CDMA Communications in Dispersive Channels,” *IEEE Trans. Commun.*, vol. 46, pp. 91–103, January.

Wang, X., and Poor, H. V. (1998b). “Blind Multiuser Detection: A Subspace Approach,” *IEEE Trans. Inform. Theory*, vol. 44, pp. 91–103, January.

Wang, X., and Poor, H. V. (1999). “Iterative (Turbo) Soft Interference Cancellation and Decoding for Coded CDMA,” *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July.

Wang, X., and Poor, H. V. (2004). *Wireless Communication Systems*, Prentice-Hall, Upper Saddle River, NJ.

Wang, X., and Wicker, S. B. (1996). “A Soft-Output Decoding Algorithm for Concatenated Systems,” *IEEE Trans. Inform. Theory*, vol. 42, pp. 543–553, March.

Ward, R. B. (1965). “Acquisition of Pseudonoise Signals by Sequential Estimation,” *IEEE Trans. Commun. Tech.*, vol. COM-13, pp. 474–483, December.

Ward, R. B., and Yiu, K. P. (1977). “Acquisition of Pseudonoise Signals by Recursion-Aided Sequential Estimation,” *IEEE Trans. Commun.*, vol. COM-25, pp. 784–794, August.

Weber, W. J., III, Stanton, P. H., and Sumida, J. T. (1978). "A Bandwidth Compressive Modulation System Using Multi-Amplitude Minimum-Shift Keying (MAMSK)," *IEEE Trans. Commun.*, vol. COM-26, pp. 543–551, May.

Wei, L. F. (1984a). "Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space, Part I: 180°," *IEEE J. Selected Areas Commun.*, vol. SAC-2, pp. 659–671, September.

Wei, L. F. (1984b). "Rotationally Invariant Convolutional Channel Coding with Expanded Signal Space, Part II: Nonlinear Codes," *IEEE J. Selected Areas Commun.*, vol. SAC-2, pp. 672–686, September.

Wei, L. F. (1987). "Trellis-Coded Modulation with Multi-Dimensional Constellations," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 483–501, July.

Weingarten, H., Steinberg, Y., and Shamai, S. (2004). "The Capacity Region of the Gaussian MIMO Broadcast Channel," *Proc. Conf. Inform. Sci. Syst. (CISS)*, pp. 7–12, Princeton, NJ, March.

Weinstein, S. B., and Ebert, P. M. (1971). "Data Transmission by Frequency-Division Multiplexing Using the Discrete Fourier Transform," *IEEE Trans. Commun.*, vol. COM-19, pp. 628–634, October.

Welch, L. R. (1974). "Lower Bounds on the Maximum Cross Correlation of Signals," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 397–399, May.

Weldon, E. J., Jr. (1971). "'Decoding Binary Block Codes on Q-ary Output Channels," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 713–718, November.

Werner, J. J. (1991). "The HDSL Environment," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 785–800, August.

Wesolowski, K. (1987a). "An Efficient DFE and ML Suboptimum Receiver for Data Transmission over Dispersive Channels Using Two-Dimensional Signal Constellations," *IEEE Trans. Commun.*, vol. COM-35, pp. 336–339, March.

Wesolowski, K. (1987b). "Efficient Digital Receiver Structure for Trellis-Coded Signals Transmitted Through Channels with Intersymbol Interference," *Electronics Lett.*, pp. 1265–1267, November.

Wiberg, N. (1996). "Codes and Decoding on General Graphs," Ph.D. Thesis, Linköping University, S-581 83 Linköping, Sweden.

Wiberg, N., Loeliger, H. A., and Kötter, R. (1995). "Codes and Iterative Decoding on General Graphs," *European Trans. Telecomm.*, vol. 6, pp. 513–525.

Wicker, S. B. (1995). *Error Control Systems for Digital Communication and Storage*, Prentice-Hall, Upper Saddle River, NJ.

Wicker, S. B., and Bhargava, V. K. (1994). *Reed Solomon Codes and their Applications*, IEEE Press, New York.

Widrow, B. (1966). "Adaptive Filters, I: Fundamentals," Stanford Electronics Laboratory, Stanford University, Stanford, CA, Tech Report No. 6764-6, December.

Widrow, B. (1970). "Adaptive Filters," in *Aspects of Network and System Theory*, R. E. Kalman and N. DeClaris (eds.), Holt, Rinehart and Winston, New York.

Wiener, N. (1949). *The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications*, Wiley, New York. (Reprint of original work published as an MIT Radiation Laboratory Report in 1942.)

Wilkinson, T. A., and Jones, A. E. (1995). "Minimization of the Peak-to-Mean Envelope Power Ratio of Multicarrier Transmission Schemes by Block Coding," *Proc. IEEE Vehicular Tech. Conf.*, pp. 825–829, July.

Wilson, S. G., and Leung, Y. S. (1987). "Trellis Coded Phase Modulation on Rayleigh Channels," in *Proce. IEEE Int. Conf. Commun. (ICC)*.

Wilson, S. G., and Hall, E. K. (1998). "Design and Analysis of Turbo Codes on Rayleigh Fading Channels," *IEEE J. Selected Areas Commun.*, vol. 16, pp. 160–174, February.

Windpassinger, C., Fischer, R. F. H., and Huber, J. B. (2004b) "Lattice-Reduction-aided Broadcast Precoding," *IEEE Trans. Commun.*, vol. 52, pp. 2057–2060, December.

Windpassinger, C., Fischer, R. F. H., Vencel, T., and Huber, J. B. (2004a) "Precoding in Multi-antenna and Multi-user Communications," *IEEE Trans. Wireless Commun.*, vol. 3, pp. 1305–1366, July.

Windpassinger, C., Vencel, T., and Fischer, R. F. H. (2003). "Precoding and Loading for BLAST-like Systems," *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 5, pp. 3061–3065, Anchorage, AK, May.

Winters, J. H., Salz, J., and Gitlin, R. D. (1994). "The Impact of Antenna Diversity on the Capacity of Wireless Communication Systems," *IEEE Trans. Commun.*, vol. COM-42, pp. 1740–1751, Feb./March/April.

Wintz, P. A. (1972). "Transform Picture Coding," *Proc. IEEE*, vol. 60, pp. 880–920, July.

Wittneben, A. (1993). "A New Bandwidth Efficient Antenna Modulation Diversity Scheme for Linear Digital Modulation," *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 3, pp. 1630–1634.

Wolf, J. K. (1978). "Efficient Maximum Likelihood Decoding of Linear Block Codes Using a Trellis," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 76–81, January.

Wolfowitz, J. (1978). *Coding Theorems of Information Theory*, 3d ed., Springer-Verlag, New York.

Wozencraft, J. M. (1957). "Sequential Decoding for Reliable Communication," *IRE Nat. Conv. Rec.*, vol. 5, pt. 2, pp. 11–25.

Wozencraft, J. M., and Jacobs, I. M. (1965). *Principles of Communication Engineering*, Wiley, New York.

Wozencraft, J. M., and Kennedy, R. S. (1966). "Modulation and Demodulation for Probabilistic Decoding," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 291–297, July.

Wozencraft, J. M., and Reiffen, B. (1961). *Sequential Decoding*, MIT Press, Cambridge, MA.

Wulich, D. (1996). "Reduction of Peak-to-Mean Ratio of Multicarrier Modulation Using Cyclic Coding," *Electr. Lett.*, vol. 32, pp. 432–433, February.

Wulich, D., and Goldfeld, L. (1999). "Reduction of Peak Factor in Orthogonal Multicarrier Modulation by Amplitude Limiting and Coding," *IEEE Trans. Commun.*, vol. 47, pp. 18–21, January.

Wunder, G., and Boche, H. (2003). "Upper Bounds on the Statistical Distrubution of the Crest-Factor in OFDM Transmission," *IEEE Trans. Inform. Theory*, vol. 49, pp. 488–494, February.

Wyner, A. D. (1965). "Capacity of the Band-Limited Gaussian Channel," *Bell. Syst. Tech. J.*, vol. 45, pp. 359–371, March.

Xie, Z., Rushforth, C. K., and Short, R. T. (1990a). "Multiuser Signal Detection Using Sequential Decoding," *IEEE Trans. Commun.*, vol. COM-38, pp. 578–583, May.

Xie, Z., Short, R. T., and Rushforth, C. K. (1990b). "A Family of Suboptimum Detectors for Coherent Multiuser Communications," *IEEE J. Selected Areas Commun.*, vol. SAC-8, pp. 683–690, May.

Yao, H., and Wornell, G. W. (2002). "Lattice-reduction-aided Detectors for MIMO Communication Systems," *Proc. 2002 IEEE Global Telecommunications Conf. (GLOBECOM)*, vol. 1, pp. 424–428, November.

Yao, K. (1972). "On Minimum Average Probability of Error Expression for Binary Pulse-Communication System with Intersymbol Interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 528–531, July.

Yao, K., and Tobin, R. M. (1976). "Moment Space Upper and Lower Error Bounds for Digital Systems with Intersymbol Interference," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 65–74, January.

Yasuda, Y., Kashiki, K., and Hirata, Y. (1984). "High-Rate Punctured Convolutional Codes for Soft-Decision Viterbi Decoding," *IEEE Trans. Commun.*, vol. COM-32, pp. 315–319, March.

Yu, W., and Cioffi, J. (2002). "Trellis Precoding for the Broadcast Channel," *Proc. GLOBECOM Conf.*, pp. 1344–1348. October.

Yu, W., and Cioffi, J. (2001). "Sum Capacity of a Gaussian Vector Broadcast Channel," *Proc. IEEE Int. Symp. Inform. Theory*, p. 498, July.

Yue, O. (1983). "Spread Spectrum Mobile Radio 1977–1982," *IEEE Trans. Vehicular Tech.*, vol. VT-32, pp. 98–105, February.

Zehavi, E. (1992). "8-PSK Trellis Codes for a Rayleigh Channel," *IEEE Trans. Commun.*, vol. 40, pp. 873–884, May.

Zelinski, P., and Noll, P. (1977). "Adaptive Transform Coding of Speech Signals," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. ASSP-25, pp. 299–309, August.

Zervas, E., Proakis, J. G., and Eyuboglu, V. (1991). "A Quantized Channel Approach to Blind Equalization," *Proc. ICC'91*, Chicago, IL, June.

Zhang, J-K., Kavcic, A., and Wong, K. M. (2005). "Equal-Diagonal QR Decomposition and Its Application to Precoder Design for Successive-Cancellation-Detection," *IEEE Trans. Inform. Theory*, vol. 51, pp. 154–172, January.

Zhang, X., and Brady, D. (1993). "Soft-Decision Multistage Detection of Asynchronous AWGN Channels," *Proc. 31st Allerton Conf. on Commun., Contr., Comp.* Allerton, IL, October.

Zhou, K., and Proakis, J. G. (1988). "Coded Reduced-Bandwidth QAM with Decision-Feedback Equalization," *Conf. Rec. IEEE Int. Conf. Commun.*, Philadelphia, PA, pp. 12.6.1–12.6.5, June.

Zhou, K., Proakis, J. G., and Ling, F. (1990). "Decision-Feedback Equalization of Time-Dispersive Channels with Coded Modulation," *IEEE Trans. Commun.*, vol. COM-38, pp. 18–24 January.

Zhu, X., and Murch, R. D. (2002). "Performance Analysis of Maximum Likelihood Detection in a MIMO Antenna System," *IEEE Trans. Commun.*, vol. 50, pp. 187–191, February.

Zigangirov, K. S. (1966). "Some Sequential Decoding Procedures," *Probl. Peredach. Inform.*, vol. 2, pp. 13–25.

Ziv, J. (1985). "Universal Quantization," *IEEE Trans. Inform. Theory*, vol. 31, pp. 344–347.

Ziv, J., and Lempel, A. (1977). "A Universal Algorithm for Sequential Data Compression," *IEEE Trans. Inform. Theory*, vol. IT-23, pp. 337–343.

Ziv, J., and Lempel, A. (1978). "Compression of Individual Sequences via Variable-Rate Coding," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 530–536.

Zvonar, Z., and Brady, D. (1995). "Differentially Coherent Multiuser Detection in Asynchronous CDMA Flat Rayleigh Fading Channels," *IEEE Trans. Commun.*, vol. COM-43, pp. 1252–1255, February/March/April.

INDEX

a posteriori
  L-values, 545
  probabilities, 162
a priori
  L-values, 552
  probabilities, 162
Abelian group, 403
  cyclic subgroup, 482
Adaptive equalization, 689
Adaptive equalizers, (*See also* Equalizers), 689–731
  accelerating convergence of LMS, 700–701
  blind, 721–731
  decision-feedback, 705–706
  fractionally-spaced, 702–703
  linear, 689–702
  maximum likelihood sequence estimator, 703–705, 721–725
  reduced state, 708–710
Affine transformation, 66
Alamouti code, 1007–1011
Algorithm
  BCJR, 541
  belief propagation, 570
  Berlekamp-Massey, 469
  constant-modulus, 726-730
  FFT, 749–752
  Godard, 726–730
  Levinson-Durbin, 692, 716
  LLL, 1067
  LMS (MSE), 691–693
  recursive least-squares (RLS), 710–714
  RLS (fast), 715
  RLS (Kalman), 711–714
  RLS lattice, 718
  RLS square-root, 715
  soft-output Viterbi algorithm (SOVA), 532
  stochastic gradient, 691–693, 725–730
  sum-product, 558
  tap-leakage, 702
  Viterbi, 243–246, 510–513
  zero-forcing, 690–691
Aliasing, 75
ALOHA protocols, 1069–1073
  slotted, 1070
  unslotted, 1070
ALOHA systems, 1069–1073
  throughput, 1071–1073
Amplitude distortion, 598
Amplitude-shift keying (ASK), 99
Analytic signal, 21
Antenna
  beamwidth, 263
  effective area, 262
  effective radiated power, 262
  illumination efficiency factor, 262
  multiple antenna systems, 996–1021
Antipodal signals, 101
ARQ (automatic repeat request), 432
ASK, 99
  error probability, 189
Asymmetric digital subscriber line (ADSL), 756
Asymptotic coding gain, 426
Augmented codes, 447
Autocorrelation function, 67
  for in-phase component, 80
  for lowpass process, 81
  for quadrature component, 80
Automatic gain control (AGC), 294
Automatic repeat request (ARQ), 432
Average energy per bit, 97
Average signal energy, 97
AWGN channel model, 10

Backward recursion, 543
Band-limited channels (*See also* Channels), 597–598
  characterization of, 598–601
Bandlimited random processes, 75
Bandpass processes, 79
  in-phase component, 79
  lowpass equivalent, 79
  quadrature component, 79
Bandpass signal, 21
Bandwidth efficiency, 226
Bandwidth expansion factor, 428
Bandwidth of a signal, 20
Bandwidth of complex signals, 20
Baseband signal,
  NRZ, 115
  NRZI, 115
Baseline figure of merit, 239
Baudot code, 12
BCH codes, 463
  Berlekamp-Massey algorithm, 469
  decoding, 467
  error location numbers, 468
  error locator polynomial, 468
  generator polynomial, 464
  non-binary, 471
  syndrome, 467
BCJR algorithm, 541
  backward recursion, 543
  forward recursion, 543
  SISO decoder, 545
  soft output, 544
Belief propagation algorithm, 570
Berlekamp-Massey algorithm, 469
Bernoulli random variable, 40
Bessel function
  modified, 47, 213
BEXPERM, 950–951
Bhatacharyya parameter, 373
  for binary input channel, 376
Bias term, 171
Bibliography, 1109
BICM (bit-interleaved coded modulation), 936
Binary antipodal signaling, 101
  error probability, 174
  optimal detection, 173
Binary entropy function, 334
Binary equiprobable signaling
  error probability, 174
Binary expurgated permutation modulation (BEXPERM), 950–951
Binary FSK
  error probability for noncoherent detection, 218
Binary orthogonal signaling,
  error probability, 176
  optimal detection, 176
Binary modulation, 2
Binary PSK (BPSK), 102
Binary Symmetric Channel (BSC), 355
Binomial random variable, 41
Biorthogonal signaling, 111
  error probability, 208
  optimal detection, 207
Bipartite graph, 559
Bit, 2
Bit error probability, 164, 417
  BPSK, 192
  PSK, 197
Bit interval, 97
Bit rate, 97
Bit-interleaved coded modulation (BICM), 936
Blind equalization, 721–731
  constant modulus algorithm, 726–730
  Godard algorithm, 726–730
  joint data and channel estimation, 724–725
  maximum-likelihood algorithms, 721–725
  stochastic gradient algorithms, 725–730
  with second-order moments, 730–731
Block error probability, 417
Block interleaver, 476
Boltzmann's constant, 69
Bounds
  Chernov, 58, 373, 866–868, 923
  Elias, 443
  Hamming, 441
  McEliece-Rodemich-Rumsey-Welch (MRRW), 443
  Plotkin, 442
  Singleton, 440
  sphere packing, 441
  Varshamov-Gilbert, 443
  Welch, 801
BPSK, 102
  bit error probability, 192
Broadcast channels, 1053–1068
  linear precoding for, 1055–1058
    MMSE, 1057
    ZF, 1057
  nonlinear precoding for, 1058–1068
    lattice reduction, 1065–1068
    QR decomposition, 1058–1062
    vector precoding, 1062–1065
BSC (binary symmetric channel), 355
Burst error correcting codes, 475–477
  Burton codes, 475
  Fire codes, 475
  Reed-Solomon codes, 471–475
Burst of errors, 475
Burton codes, 475

Capacity, 13, 360
  ε-outage, 907
  achieved by orthogonal signaling, 367
  bandlimited AWGN channel, 365
  discrete-time AWGN channel, 365
  discrete-time binary-input channel, 362
  ergodic, 900
    of MIMO channels, 985–986, 990–991
  finite-state channels, 903

  - of MIMO channels, 981–991
  - of multicarrier system, 744–745
  - of multiple access methods, 1031–1035
  - outage, 987–990
  - symmetric channels, 363
- Carrier phase estimation, 292–298
  - Costas loop, 312–313
  - decision-directed, 303–308
  - for multi-phase signals, 313–314
  - ML methods, 296–298, 321–322
  - nondecision directed, 308–315
  - phase-locked loop, 298–303
  - squaring loop, 310–312
- Carrier recovery, 290–295
- Carrier sense multiple access (CSMA), 1073
  - protocols, 1074–1077
  - nonpersistent, 1074
  - 1-persistent, 1074
  - *p*-persistent, 1074–1077
- Catastrophic convolutional codes, 509
- Cauchy-Schwarz inequality, 29–30
- Central frequency, 21
- Central limit theorem (CLT), 63
- CFM (constellation figure of merit), 238
- Chain rule for entropies, 335
- Channel
  - access protocol, 1069
  - acoustic, 9
  - additive noise, 10
  - additive Gaussian noise, 10
  - AWGN, 160
  - band-limited, 597–598
  - binary symmetric (BSC), 355
  - broadcast, 1053–1068
  - capacity, 13, 360
    - of MIMO channels, 981–991
  - coherence bandwidth, 835
  - coherence time, 836
  - cutoff rate (R_0), 527, 787–791
    - for fading channels, 957–960
  - discrete-input continuous-output, 357
  - discrete-memoryless, 356
  - discrete-time AWGN, 358
  - discrete-time model, 625–628
  - distortion, 598–601
    - amplitude, 598
    - envelope delay, 598–599
    - frequency offset, 600
    - impulse noise, 601
    - nonlinear, 600
    - peak, 641
    - phase jitter, 600
    - squared-error, 645–646
    - thermal noise, 600
  - Doppler power spectrum, 836
  - Doppler spread, 836
  - encoder, 1
    - code rate, 2, 402
    - codeword, 2, 372, 401
  - envelope delay, 598–599
  - fading multipath, characterization of, 831–833
    - correlation functions for, 833–839
    - impulse response, 832
    - models for, 839–843
    - transfer function, 834
  - fiber optic, 4
  - finite-state, 903
  - frequency nonselective, 836, 844
    - digital signaling over, 844
  - frequency selective, 836, 844
    - digital signaling over, 869–889
    - error rate for, 872–880
    - RAKE demodulator for, 871–872
    - tap weight estimation of, 876–877
    - tapped delay line model of, 869–871
  - frequency offset, 600
  - impulse noise, 601
  - memoryless, 355
  - microwave LOS, 8
  - models for,
    - additive noise, 10
    - binary symmetric, 355
    - COST 207, 840
    - discrete memoryless, 356
    - discrete-time, 358
    - for multiuser channels, 1037–1038
    - Hata, 843
    - Jakes' model, 838–839
    - linear filter, 11
    - linear, time-variant filter, 11–12, 832
    - MIMO channels, 966
    - slowly fading, 845
    - statistical, 839–843
    - waveform, 358
  - multipath spread, 834
  - Nakagami fading, 841
  - nonlinear, 600
  - overspread, 845
  - phase jitter, 600
  - probability transition matrix, 357
  - Rayleigh fading, 833
    - Binary signaling over, 847–849
    - coded waveforms for, 942–956
    - coding for, 899–960
    - cutoff rate for, 957–960
    - frequency nonselective, 846–849
    - *M*-ary orthogonal signaling over, 861–865
    - Multiphase signaling over, 859–861
  - reliability function, 369
  - state information (CSI), 904, 957–960, 1054
  - Ricean fading, 833
  - scattering function, 837
  - spread factor, 845
    - table, 845
  - squared-error, 645–646
  - storage, 9
  - symmetric, 363
  - thermal noise, 3, 69, 600
  - throughput, 1070
  - underspread, 845
  - underwater acoustic, 9
  - waveform, 358
  - wireless, 5–9
  - wireline, 4
- Channel capacity, 13, 360
- Channel coding, 400
- Channel *L*-value, 552
- Channel state information (CSI), 904, 957–960, 1054
- Characteristic function, 44
- Characteristic of a field, 404
- Chernov bound, 58, 373, 923
  - for Rayleigh fading channel, 866–868
  - pairwise error probability, 1014–1016
- Chernov parameter, 373
- χ^2 random variable, 45
- Circular random vectors, 66
- Clairvoyant estimate, 1098
- CLT (central limit theorem), 63
- Code division multiple access (CDMA), 780–784
  - asymptotic efficiency, 1052
  - asynchronous, 1039–1042
  - capacity of, 1033–1034
  - digital cellular, 780–784
  - frequency hopped, 802–804, 813–814
  - optimum receiver for, 1038–1042
  - suboptimum detectors for, 1042–1050
    - decorrelating, 1043–1045
    - MMSE, 1046–1047
    - multistage interference cancellation, 1048–1049
    - performance, 1050–1053
    - single user, 1042–1043
    - successive interference cancellation, 1047–1048
  - synchronous, 1038–1039
- Code rate, 2
- Codeword, 2
- Coded modulation,
  - bit-interleaved, 936
  - trellis, 571–586, 929–935
- Codes
  - augmented, 447
  - bandwidth efficient, 571, 586
  - bandwidth expansion factor, 428
  - BCH, 463
  - bit error probability, 417
  - block, 401
  - block error probability, 417
  - burst error correcting, 475
  - Burton, 475
  - classification, 401
  - coding gain, 426, 533
  - concatenated, 479–480, 953–956, 1020–1021
  - conditional weight enumeration function, 416
  - constant weight, 949–953
  - convolutional, 491–548, 946–948
  - coset, 430
  - CRC, 453
  - cyclic, 447
  - cyclic Golay, 460
  - cyclic Hamming, 460
  - diversity order, 927
  - dual, 412
  - effective distance, 927
  - equivalent, 412
  - expurgated, 447, 950–951
  - extended, 447
  - extended Golay, 424
  - Fire, 475
  - fixed weight, 411, 949–953
  - generator matrix, 412
  - Golay, 424, 460
  - Hadamard, 423, 951–953
  - Hamming, 420, 460
  - Hamming distance, 414
  - inner, 479
  - input-output weight enumeration function, 416
  - instantaneous, 340
  - lengthened, 446
  - linear block, 411
  - low density parity check (LDPC), 569
  - maximum distance separable, 440
  - maximum length, 421
  - maximum-length shift register, 461
  - MDS (maximum-distance separable), 440
  - minimum distance, 414
  - minimum weight, 414
  - outer, 479
  - parallel concatenated block, 481
  - parity check matrix, 412
  - perfect, 434, 442
  - product, 477
  - punctured, 446, 516–517, 521–523
  - quasi-perfect, 435
  - rate, 2
  - Reed-Muller (RM), 421
  - Reed-Solomon (RS), 471
  - serially concatenated block, 480
  - shortened, 445
  - shortened cyclic, 452
  - standard array, 430
  - syndrome, 430, 467
  - systematic, 412
  - ternary Golay, 442
  - turbo, 548
  - undetected error, 430
  - uniquely decodable, 339
  - weight distribution, 411
  - weight distribution polynomial, 415

- Codes (*continued*)
  - weight enumeration function, 415
  - word error probability, 417
- Codeword, 372, 401
  - weight, 411
- Coding
  - diversity order, 927
  - effective distance, 927
  - for MIMO channels, 1001–1021
  - for Rayleigh fading channel, 942–960
    - concatenated, 953–956
    - constant-weight codes, 949–953
    - convolutional codes, 946–948
    - cutoff rate, 371–380, 516, 527, 787–791, 957–960
    - linear block codes, 943–946
    - space-time codes, 1006–1021
    - trellis codes, 1016–1019
  - Gray, 100
  - Huffman, 342–346
  - in the frequency domain, 942–960
- Coding gain, 533
  - of a lattice, 233
- Complementary error function, 44
- Complementary gamma function, 911
- Complete set of signals, 32
- Complex envelope, 22
- Complex random processes
  - covariance, 71
  - pseudocovariance, 71
- Complex random variables, 63
- Complex random vectors, 64
  - covariance matrix, 64
  - pseudocovariance matrix, 64
- Complex signals
  - bandwidth, 20
- Concatenated codes, 479–480, 540–541, 953–956, 1020–1021
  - inner code, 479, 540
  - outer code, 479, 540
- Concave function, 386
- Conditional entropy, 334
- Conditional weight enumeration function, 416
- Confluent hypergeometric function, 49
- Conjugacy class, 409
- Conjugate element, 409
- Constant weight codes, 411, 949–953
- Constellation, 34
  - figure of merit, 238
  - minimum distance, 185
- Constellation figure of merit (CFM), 238
- Constraint length, 96, 491
- Continuous-phase frequency-shift keying (CPFSK), 116–118
  - performance of, 116
  - power density spectrum of, 138–145
  - representation of, 116–117
- Continuous-phase modulation (CPM), 118–123, 243–259
  - demodulation, 243–258
    - maximum-likelihood sequence estimation, 243–246
    - metric computations, 249–251
    - multi-*h*, 257–258
    - performance of, 251–258
    - suboptimum, 258–259
  - full response, 118
  - linear representation of, 128–130
  - minimum-shift keying (MSK), 123–124
  - modulation index, 118, 254
  - multi-*h*, 118, 257–258
  - partial response, 118
  - phase cylinder, 122
  - phase state, 248
  - phase trees of, 120
  - power spectrum of, 138–142, 145–148
  - representation of, 118–123
  - state trellis, 249
  - trellis of, 120
- Continuous-wave (CW) interference, 772
- Convergence
  - almost everywhere (a.e.), 63
  - almost surely (a.s), 63
  - in distribution, 63
- Convex functions, 386
- Convolutional codes, 491–548
  - applications, 532–537
  - catastrophic, 509
  - constraint length, 491
  - concatenated, 540–541
  - decoding,
    - Fano algorithm, 525
    - feedback, 529–531
    - maximum a posteriori, 541–548
    - sequential, 525–528
    - stack algorithm, 528–529
    - Viterbi, 243–246
  - distance properties of, 516
  - dual-*k*, 537–540
  - equivalent encoders, 506
  - first-event error, 502
  - first-event error probability, 513
  - hard-decision decoding, 945–946
  - invertibility conditions, 508
  - invertible, 508
  - maximum free distance, 516
  - nonbinary, 499, 504
  - parallel concatenated (PCCC), 548
  - performance on AGWN channel, 513–516
  - performance on BSC, 513–516
  - performance on Rayleigh fading channel, 946–948
  - punctured, 516–517, 521–523
  - rate, 491
  - rate-compatible punctured, 523–525
  - recursive systematic (RSCC), 507–508
  - soft-decision decoding, 943–944
  - state diagram, 496
  - systematic, 505
  - table of generators for maximum free distance, 517–520
  - transfer function, 500
  - tree diagram, 496
  - trellis diagram, 496
  - Viterbi algorithm, 510
- Convolutional interleavers, 476
- Correlation metric, 173
- Correlation receiver, 177
- Correlative state, 248
- Correlative state vector, 248
- Coset, 430, 483
- Coset leader, 430
- Coset representative, 584
- Covariance
  - for complex random processes, 71
- CPFSK, 116–118, 138–145
  - modulation index, 118
  - peak frequency deviation, 117
  - power spectral density, 138–145
- CPM, (See Continuous-Phase Modulation),
- CRC codes, 453
- Cross spectral density, 67
  - in-phase and quadrature components, 80
- Cross-correlation coefficient, 26
- Crosscorrelation function, 67
  - in-phase and quadrature components, 80
- CSD (cross spectral density), 67
- CSI (channel state information), 904, 957–960, 1054
- Cutoff rate (R_0), 371–380, 516, 527
  - comparison with channel capacity, 377–380
  - for fading channels, 957–960
  - for pulsed interference, 787–791
- CWEF (conditional weight enumeration function), 416
- Cyclic codes, 447
  - CRC, 453
  - decoding, 458
  - encoding, 455
  - generator polynomial, 448
  - Golay, 460
  - Hamming, 460
  - message polynomial, 449
  - parity check polynomial, 450
  - shortened, 452
  - systematic, 453
- Cyclic equalization, 694
- Cyclic redundancy check (CRC) codes, 453
- Cyclic subgroup, 482
- Cyclostationary random process, 70
- *D* transform, 493
- Data compression, 1, 335–354
  - lossless, 335–348
  - lossy, 348–354
- Decision-feedback equalizer (*see* Equalizers, decision-feedback), 661–665, 705–706
- Decision region, 163
- Decoding,
  - Berlekamp-Massey, 469
  - Fano algorithm, 525
  - feedback, 529–531
  - hard decision, 428
  - iterative, 478, 548
  - Meggit, 460
  - sequential, 525–528
  - soft decision, 424
  - stack algorithm, 528–529
  - turbo, 552
  - LDPC, 570
  - Viterbi algorithm, 243–244,
- Degrees of freedom, 75
- Delay distortion, 598–599
- Delay power spectrum, 834
- Demodulation, 24
- Demodulation and detection, 201
  - carrier recovery for, (See Carrier phase estimation)
  - coherent
    - comparison of, 226–229
    - of binary signals, 173–177
    - of biorthogonal signals, 207–209
    - of orthogonal signal, 203–207
    - of PAM signals, 188–190
    - of PSK signals, 190–195
    - of QAM signals, 196–200
    - optimum, 201–203
  - correlation type, 177–178
  - of CPM, 243–258
    - performance, 251–258
  - for intersymbol interference, 623–628
  - matched filter-type, 178–182
  - maximum likelihood, 163
  - maximum-likelihood sequence, 623–628
  - noncoherent, 210–224
    - of binary signals, 219–221
    - of *M*-ary orthogonal signals, 216–219, 741–743, 861–865
    - multichannel,737–743
    - optimum, 212–214
  - of OFDM, 749
- Density of a lattice, 236
- Detector
  - decorrelating, 1043–1045
  - envelope, 214
  - inverse channel (ICD), 970
  - maximum-likelihood (MLD), 970
  - MMSE, 970, 1046–1047
  - minimum distance, 171
  - nearest neighbor, 171
  - nonlinear, 973–974
  - optimal noncoherent, 212–214

  single user, 1042–1043
  sphere, 973
Differential encoding, 115
Differential entropy, 349
Differential phase-shift keying (DPSK), 221
Differentially encoded PSK, 195
Digamma function, 909
Digital communication system model, 1–3
Digital modulation, 95
Digital modulator, 2
Digital signaling, 95
Dimensionality theorem, 227
Direct sequence (*See* Spread spectrum signals)
Dirty paper precoding, 1054
Discrete memoryless source (DMS), 331
Discrete-memoryless channel (DMC), 356
Discrete-time AWGN, 358
Discrete-time AWGN channel capacity, 365
Discrete-time binary-input channel capacity, 362
Distance (*see* Block codes, Convolutional codes)
  effective, 927
  enumerator function, 185
  Euclidean, 35
  Hamming, 414
  metric, 173
  product, 925
Distortion (*see* Channel distortion)
  Hamming, 354
  squared-error, 350
Distortion-rate function, 352
Diversity
  antenna, 851
  frequency, 850
  gain, 996–997
  order, 852, 927
  performance of, 851–859
  polarization, 851
  RAKE, 851
  signal space, 928
  time, 851
DMC (*see* Discret Memoryless Channel)
DMS (*see* Discret Memoryless Source)
Double-sideband (DSB) PAM, 100
DPSK, 221
  error probability, 223
DSB, 100
Dual code, 412
Dual-k codes, 537–540
Duobinary signal, 610

ε-outage capacity, 907
Early-late gate synchronizer, 318–321
Effective antenna area, 262
Effective distance, 927
Effective radiated power, 260–261
Eigenvalue, 29, 1086
Eigenvector, 29, 1086
Elias bound, 443
Encoder
  catastrophic, 509
  convolutional, 402, 492
  for cyclic codes, 455
  inverse, 508
  turbo, 549
Encoding (*see* Block codes; Convolutional codes)
Energy, 25
  average, 97
  per bit,
    average, 97
Entropy, 333
  chain rule, 335
  conditional, 334
  differential, 349
  joint, 334
Entropy rate, 337
Envelope detection, 214
Envelope of a signal, 23
Equivalent codes, 412
Equivalent convolutional encoders, 506
Equalizers (*See also* Adaptive equalizers)
  at transmitter, 668–669
  decision-feedback, 661–665, 705–706
    adaptive, 689–731
    examples of performance, 662–665
    for MIMO channels, 979–981
    of trellis-coded signals, 706–708
    minimum MSE, 663
    predictive form, 665–667
  linear, 640–649
    adaptive, 689–693
    baseband, 658–659
    convergence of MSE algorithm, 695–696
    cyclic equalization, 694
    error probability, 651–655
    examples of performance, 651–655
    excess MSE, 696–697
    for MIMO channels, 975–979
    fractionally spaced, 655–658
    LMS (MSE) algorithm, 691–693
    mean-square error (MSE) criterion, 645–655
    minimum MSE, 647–648
    output SNR for, 648
    passband, 658–659
    peak distortion, 641
    peak distortion criterion, 641–645
    phase-splitting, 659
    zero-forcing, 642
  iterative equalization/decoding, 671–673
  maximum a posteriority probability (MAP), 291
  maximum –likelihood sequence estimation, 623–625,
    reduced-state, 669–671
  self-recovering (blind), 721–731
  with trellis-coded modulation, 706–708
  using the Viterbi algorithm, 628–631
    channel estimator for, 703–705
    performance of, 631–639
    reduced complexity, 669–671
    reduced-state, 669–671
erfc, 44
Ergodic capacity, 900, 905–906, 985–987
Error correction, 900
Error detection, 432
Error floor, 551
Error probability,
  16QAM, 186, 200
  ASK, 189
  binary antipodal signaling, 174
  binary equiprobable signaling, 174
  binary orthogonal signaling, 176
  biorthogonal signaling, 208
  bit, 164, 417
  block, 417
  DPSK, 223
  for hard-decision decoding, 945–946
  for soft-decision decoding, 943–944
  FSK, 205
  lower bound to, 186
  M-ary PSK, 190–194
    for Rayleigh fading, 859–861, 1100–1103
    for Ricean fading, 1104–1105
    for AWGN channel, 1106
  message, 164
  multichannel binary symbols, 739–741, 1090–1095
  orthogonal signaling, 205
    noncoherent detection, 216
  pairwise, 184, 372, 418, 922, 928
  PAM, 189
  QAM, 198
  QPSK, 199
  symbol, 164
  union bound, 182
  word, 417
Estimate
  biased, 323
  clairvoyant, 1098
  consistent, 324
  efficient, 324
  pilot signal, 1098
  unbiased, 323
Estimate of phase (*See* Carrier phase estimation)
Estimation
  maximum-likelihood, 291, 296–298, 321–322
  of carrier phase, 295–315
  of signal parameters, 290
  of symbol timing, 290
  of symbol timing and carrier phase, 321–322
  performance of, 323–326
Euclidean distance, 35
Euler's constant, 909
Excess bandwidth, 607
Excess MSE, 696–697
Excision of narrowband interference, 791–796
  linear, 792–796
  nonlinear, 796
EXIT charts, 555
Exponential random variable, 46
Expurgated codes, 447, 950–951
Extended codes, 447
Extended Golay code, 424
Extension field, 404
Extrinsic information, 552
Extrinsic L-value, 552
Eye pattern, 603

Factor Graphs, 558
Fading, 8, 830–844
  figure, 52
Fading channels (*See also* Channels), 830–890
  coding for, 899–960
  ergodic capacity, 900, 905–906, 985–987
  outage capacity, 900, 906, 907, 900, 987–990
  propagation models for, 842–843
Feedback decoding, 529–531
FH spread spectrum signals (*see* Spread spectrum signals),
Field
  characteristic, 404
  extension, 404
  finite, 403
  Galois, 403
  ground, 404
  minimal polynomial of an element, 408
  order of an element, 407
  primitive element, 407
Figure of merit
  baseline, 239
  constellation, 238
Filtered multitone (FMT) modulation, 754
Filters,
  matched, 178–182
  whitening, 627
Finite fields, 403
Finite-state channels, 903
  capacity, 903–905
Fire codes, 475
First-event error, 502
First-event error probability, 513
Fixed weight codes, 411, 949–953
Fixed-length source coding, 339

- Folded spectrum, 644
- Forward recursion, 543
- Free Euclidian distance, 577
- Free-space path loss, 262
- Frequency diversity, 850
- Frequency range
  - wireline channels, 5
  - wireless (radio) channels, 6
- Frequency division multiple access (FDMA), 1029
  - capacity of,1031–1032
- Frequency domain coding, 942–960
- Frequency hopped (FH) spread spectrum, 802–804
- Frequency support, 20
- Frequency-shift keying (FSK), 109–110
  - continuous-phase (CPFSK), 116–118
  - error probability, 205
  - noncoherent detection, 215
  - power density spectrum, 154
- Frobenius norm, 982
- Fundamental coding gain, 586
- Fundamental volume of a lattice, 233

- Galois fields, 403
  - minimal polynomial, 464
  - subfield, 483
- Gamma function, 45
  - complementary, 911
  - Digamma function, 909
- Gamma random variable, 46
- Gaussian minimum-shift keying (GMSK), 118
- Gaussian noise, 10
- Gaussian random process, 10, 68
- Gaussian random variable, 41
- Generalized RAKE demodulator, 880–882
- Generator matrix
  - lattice, 231
  - of linear block codes, 412
  - of space-time block code, 1006
  - transform domain, 495
- Generator polynomial, 448, 464
- Gilbert-Varsharmov bound, 443
- Girth of a graph, 560
- GMSK, 118, 127
- Golay codes, 424, 460
  - extended, 424
  - ternary, 442
- Gold sequences, 799
- Gram-Schmidt procedure, 29
- Graphs, 558–568
  - bipartite, 559
  - constraint nodes, 561
  - cycle-free, 560
  - cycles, 560
  - factor, 558
  - girth, 560
  - global function, 561
  - local functions, 561
  - Tanner, 558
  - variable nodes, 560
- Gray coding, 100
- Gray labeling, 939
- Ground field, 404
- Group
  - Abelian, 403
  - identity element, 404

- Hadamard codes, 423, 951–953
- Hamming bound, 441
- Hamming codes, 420, 460
- Hamming distance, 414
- Hamming distortion, 354
- Hard decision decoding,
  - of block codes, 428–436
  - of convolutional codes, 509–516
- Hata model, 843
- Hermite parameter, 233
- Hermitian matrix, 65, 1085
- Hermitian symmetry, 19
- Hermitian transpose of a matrix, 28
- Hexagonal lattice, 230
- Hilbert transform, 22
- Homogeneous Markov chains, 72
- Huffman coding, 342–346

- Identity element, 404
- iid random variables, 45
- Illumination efficiency factor, 262
- Impulse noise, 601
- Impulse response,
  - for bandpass systems, 27
- In-phase component, 22
- Inequality
  - Cauchy-Schwarz, 29–30
  - Kraft, 340
  - Markov, 56
  - triangle, 29–30
- Information sequence, 1, 401
- Information source
  - discrete memoryless, 331
  - memoryless, 331
  - stationary, 331
- Inner code, 479
- Inner product, 26, 28, 30
- Input-output weight enumeration function (IOWEF), 416
- Instantaneous codes, 340
- Interference margin, 774
- Interleaver
  - block, 476
  - convolutional, 476
  - gain 552
  - uniform, 480–481
- Interleaving, 476–477
- Intersymbol interference, 599–600, 603–604
  - controlled (*see* Partial response signals), 609–611
  - discrete-time model for, 626
  - equivalent white noise filter model, 627
  - optimum demodulator for, 623–628
- Inverse channel detector (ICD), 970
- Inverse filter, 642
- Irreducible Markov chains, 73
- Irreducible polynomial, 405
- Irregular LDPC, 570
- Irrelevant information, 166
- Iterative decoding, 478, 548–558
  - error floor, 551
  - EXIT charts, 555
  - turbo cliff region, 553
  - waterfall region, 553

- Jakes' model, 838–839
- Jensen's inequality, 386
- Joint entropy, 334
- Jointly Gaussian random variables, 54
- Jointly wide-sense stationary processes, 54

- Kalman (RLS) algorithm, 711–714
- Kalman gain vector, 712
- Karhunen-Loève expansion, 76
- Kasami sequences, 799
- Kissing number of a lattice, 232
- Kolmogorov-Wiener filter, 13
- Kraft inequality, 340

- Labeling
  - Gray, 939
  - set portioning, 939
- Lattice
  - coding gain, 233
  - coset, 584
  - density, 236
  - equivalent, 231
  - filter, 716–721
  - fundamental volume, 233
  - generator matrix, 231
  - Hermite parameter, 233
  - hexagonal, 230
  - kissing number, 232
  - minimum distance, 232
  - multidimensional, 234
  - multiplicity, 232
  - recursive least squares, 708, 715
  - Schlafli, 234
  - Sublattice, 234
  - Voronoi region, 232
- Law of large numbers (LLN), 63
- LDPC (low density parity check codes), 568–571
  - code density, 569
  - decoding, 570
  - degree distribution polynomial, 570
  - irregular, 570
  - regular, 569
  - Tanner graph, 569
- Least-squares algorithms, 710–720
- Lempel-Ziv algorithm, 346–348
- Lengthened codes, 446
- Levinson-Durbin algorithm, 692, 716
- Likelihood function, 292
- Linear block codes, 400–490
- Linear equalization (*see* Equalizers, linear)
- Linear-feedback shift-register, maximum length, 798–799
- Linear filter channel, 11
- Linear modulation, 110
- Linear prediction, 716
  - backward, 718
  - forward, 717
  - residuals, 718
- Linear time-varying channel, 11
- Linearly independent signals, 30
- Link budget analysis, 261–265
- Link margin, 246
- LLN (*see* law of large numbers)
- Log-APP (log a posteriori probability), 546
- Log-MAP (log maximum a posteriori probability), 546
- Lognormal random variable, 54
- Lossless data compression, 335
- Lossless source coding theorem, 336
- Lossy data compression, 335
- Low density parity check codes (*see* LDPC)
- Lowpass equivalent, 22
- Lowpass signal, 20
- Low probability of intercept, 778–779

- MacWilliams identity, 415
- MAP (maximum a posteriori probability), 162–163, 291
- Mapping by set partitioning, 572
- Marcum's Q-function, 47
  - generalized, 47
- M-ary modulation, 2
- Markov chains, 71–74
  - aperiodic states, 73
  - equilibrium probabilities, 73
  - ergodic, 73
  - homogeneous, 72
  - irreducible, 73
  - period of state, 73
  - state, 72
  - state probability vector, 72
  - state transition matrix, 72
  - stationary probabilities, 73
  - steady-state probabilities, 73
- Markov inequality, 57–58
- Matched filter, 178–182
  - frequency domain, 179
  - receiver, 178
- Matrix
  - condition number, 1088
  - eigenvalue, 1086
  - eigenvector, 1086
  - generator, 412–413
  - Hermitian, 65
  - Hermitian transpose, 28
  - norm, 1088
  - orthogonal, 231
  - parity check, 412–413
  - rank, 1085
  - singular values, 1087
  - skew-Hermitian, 65
  - symmetric, 1085
  - trace of, 1085
  - transpose, 28
- Max-Log-APP algorithm, 548
- Max-Log-MAP algorithm, 548

Maximal ratio combiner, 852
Maximum a posteriori probability (*see* MAP),
Maximum-distance separable codes, 440
Maximum free distance codes, 516
  tables of, 517–520
Maximum-length shift register codes, 461, 798–799
Maximum likelihood,
  parameter estimation, 290–291, 321–322
    for carrier phase, 292–298
    for joint carrier and symbol, 321–322
    for symbol timing, 315–321
    performance of, 323–324
Maximum-likelihood (ML) receiver, 163, 623–625,
Maximum likelihood sequence detection (MLSD), 623–625,
Maximum ratio combining, 852
  performance of, 851–855
McEliece-Rodemich-Rumsey-Welch (MRRW) bound, 443
MDS (maximum-distance separable) codes, 440
Mean-square error (MSE) criterion, 645–655
Meggit decoder, 460
Memoryless channel, 355
Memoryless modulation, 95
Memoryless source, 331
Mercer's theorem, 77
Message error probability, 164
  PSK, 194
  QPSK, 193
Message polynomial, 449
Metric
  correlation, 173
  distance, 173
  modified distance, 173
MGF (moment generating function), 44
Microwave LOS channel, 8
MIMO channels, 966
  capacity of, 982–984, 990–991
    ergodic, 985–986
    outage, 987–990
  coding for, 1001–1021
    bit-interleaved, 1003–1006
    space-time codes, 1006–1021
    temporal, 1003–1006
  slow fading, 968–969, 975–979
  spread spectrum signals for, 992–996
MIMO systems, 966
  detectors for, 970–974
  diversity gain for, 996–997
  error rate performance, 971–973
  lattice reduction for, 973–974
  multicode, 997–1000
  multiplexing gain for, 996–997
  outage probability, 987–988
  scrambling sequence for, 997
  singular-value decomposition for, 974–975
  spread spectrum, 992–996
Minimal polynomial, 408
Minimum distance, 414
Minimum distance detector, 171
Minimum distance of a constellation, 185
Minimum distance of a lattice, 232
Minimum weight, 414
Minimum-shift keying (MSK), 123–124
  power spectrum of, 144
ML (*see* maximum-likelihood)
MLSD, 623–625,
Modified Bessel function, 47, 213
Modified distance metric, 173
Modified duobinary signal, 610
Modulation
  binary, 2
  comparison of, 226–229
  constraint length, 96
  continuous-phase FSK (CPFSK), 116–118
    power spectrum, 138–145
  continuous-phase modulation (CPM), 118–123
  digital, 95
  DPSK, 221–223
  equicorrelated (simplex), 112–113, 209–210
  frequency-shift keying (FSK), 109–110, 205, 215–216
  linear, 110
  M-ary orthogonal, 108–111, 204–207, 216–219
  memoryless, 95
  multichannel, 737–743
  multidimensional, 108–113
  NRZ, 115
  NRZI, 115
  nonlinear, 110
  OFDM, 746–752
  offset QPSK,
  phase-shift keying (PSK), 101–103, 191–195
  pulse amplitude (PAM, ASK), 98–101, 188–190
  quadrature amplitude (QAM), 103–107, 185–187, 196–200
  with memory, 95–96
Modulator, 2, 24
  binary, 2
  digital, 95
  linear, 110
  M-ary, 2
  memoryless, 95
  nonlinear, 110
  pulse amplitude, 98–101
  quadrature amplitude, 103–107
  with memory, 95–96
Moment generating function (*see* MGF)
Monic polynomial, 405
Moore-Penrose pseudoinverse, 1088
Morse code, 12, 339
MRRW (McEliece-Rodemich-Rumsey-Welch) bound, 443
MSK, 123–124, 144
Multicarrier communications, 743–759
  capacity of, 744–745
  channel coding consideration, 759
  FFT-based system, 749–752
  Filtered multitone (FMT), 754
  OFDM, 746–742
    bit allocation, 754–757
    power allocation, 754–757
  peak-to-average ratio, 757–759
  spectral characteristics, 752–754
Multichannel communications, 737–743
  noncoherent combining loss, 741
  with binary signals, 739–741
  with *M*-ary orthogonal signals, 741–743
Multicode MIMO systems, 997–1000
Multidimensional signaling, 108
Multipath channels, 8, 831
Multipath intensity profile, 834
Multipath spread, 834
Multiple access methods, 1029–1031
  capacity of, 1031–1035
  CDMA, 1033–1034
  FDMA, 1031–1032
  random accesss, 1068–1077
  TDMA, 1032–1033
Multiple antenna systems, 966–1021
  inverse channel detector, 970
  maximum-likelihood detector, 970
  minimum MSE detector, 970
  space-time codes for, 1006–1021
    concatenated codes, 1020–1021
    differential STBC, 1014
    orthogonal STBC, 1011–1013
    quasi-orthogonal STBC, 1013
    trellis codes, 1016–1019
    turbo codes, 1020–1021
Multiplexing gain, 996–997
Multiplicity of a lattice, 232
Multistage interference cancellation, 1043–1049
Multiuser communications, 1028
  multiple access, 1029–1034
  multiuser detection, 1029–1034
  random access, 1068–1077
Multiuser detection, 1034
  decorrelating detector, 1043–1045
  for asynchronous transmission, 1039–1042
  for broadcast channels, 1053–1068
  for CDMA, 1036–1053
  for random access, 1068–1077
  for synchronous transmission, 1038–1039
  single user detector, 1042–1043
Mutual information, 332

Nakagami random variable, 52, 841
Narrowband interference, 791–796
Narrowband process, 79
Narrowband signal, 18–21
Nat, 333
Nearest neighbor detector, 171
Negative spectrum, 20
Noise,
  Gaussian, 10
  thermal, 3, 69
  white, 90
Noise equivalent bandwidth, 92
Noisy channel coding theorem, 361
Non-central χ^2 random variable, 46
Noncoherent combining loss, 741
Noncoherent detection, 210–226
  error probability for orthogonal signals, 216–218
  FSK, 215–216
Nonlinear distortion, 600
Nonlinear modulation, 110
Norm
  of a matrix, 1088
  of a signal, 30
  of a vector, 28
Normal equations, 716
Normal random variable, 41
NRZ, 115
NRZI, 115
Nyquist criterion, 604–605
Nyquist rate, 13

OFDM, 746–752, 844–890
  bit and power allocation, 754–757
  degradation due to Doppler spreading, 884–889
  FFT implementation, 749–752
  ICI suppression in, 889–890
  peak-to-average ratio, 757–759
Offset QPSK (OQPSK), 124–128
On-off keying (OOK), 267, 949
Optimal detection
  after modulation, 202
  binary antipodal signaling, 173
  binary orthogonal signaling, 176
  biorthogonal signaling, 207
  simplex signaling, 209
OQPSK, 124–128
Order of a field element, 407
Orthogonal matrix, 231
Orthogonal signaling, 108
  achieving channel capacity, 367
  error probability, 205
    with noncoherent detection, 216–218
Orthogonal signals, 26, 30

Orthogonal vectors, 28
Orthogonality principle, 646
  mean-square estimation, 646
Orthonormal
  vectors, 28
  basis, 28
  signal set, 30
Outage capacity, 900, 907, 913
  of MIMO channels, 987–990
Outage probability,
  of MIMO channels, 987–988
Outer code,

Pairwise error probability (PEP), 184, 372, 514, 922, 1014–1016
  Chernov bound, 373, 1014–1016
PAM, 98–101
Parallel contatenated block codes, 481
Parallel concatenated convolutional codes (PCCC), 548
Parity check bits, 412
Parity check matrix, 412
Parity check polynomial, 450
Partial-band interference, 804
Partial response signals, 609–611
  duobinary, 610
  error probability of, 617–618
  modified duobinary, 610
  precoding for, 613
Partial-time (pulsed), 784
Path memory truncation, 246
PCBC (parallel concatenated block codes), 481
PCCC (parallel concatenated convolutional codes), 548
Peak distortion criterion, 641–645
Peak frequency deviation, 117
Peak-to-average ratio, 757–759
PEP (*see* pairwise error probability)
Perfect codes, 434, 442
Phase of a signal, 23
Phase jitter, 600
Phase-locked loop (PLL), 298–315
  Costas, 312–313
  decision-directed, 303, 308
  loop damping factor, 299
  M-law type, 313–314
  natural frequency, 299
  non-decision-directed, 308–315
  square-law type, 310–312
Phase tree, 120
Phase trellis, 120
Phase-shift keying (PSK), 101–103
Pilot signal, 1098
Plotkin bound, 442
PN sequences, 463, 796–801
Polynomial
  irreducible, 405
  minimal, 408
  monic, 405
  prime, 405
  syndrome, 458
Positive spectrum, 20
Power efficiency, 226
Power spectral density, 67
  continuous component, 133
  CPFSK, 138–145
  discrete component, 133
  for in-phase component, 80
  for lowpass process, 81
  for quadrature component, 80
  linearly modulated signals, 133
Power spectrum, 67
Pre-envelope, 21
Precoding
  for broadcast channels, 1053–1068
    dirty paper, 1054
    linear, 1055–1058
    nonlinear, 1058–1068
    QR decomposition, 1058–1062
    vector, 1062–1065
    via lattice reduction, 1065–1068
  for spectral shaping, 133–135, 611–612
Prediction (*see* Linear prediction),
Preferred sequences, 799
Prefix condition, 340
Preprocessing, 166
Prime polynomial, 405
Primitive BCH codes, 463
Primitive element, 407
Probability distributions
  binomial, 41
  chi-square,
    central, 45–46
    noncentral, 46–48
  gamma, 46
  Gaussian, 41–45
  log normal, 54
  multivariate Gaussian, 54–56
  Nakagami, 52–53
  Rayleigh, 48–50
  Rice, 50–52
  uniform, 41
Processing gain, 773–774
Probability transition matrix of a channel, 357
Product codes, 477
Product distance, 925
Prolate spheroidal wave functions, 227
Proper random processes, 71
Proper random vectors, 65
PSD (power spectral density), 67
Pseudo-noise (PN) sequences, 796–801
  autocorrelation function, 798
  generation via shift register, 797
  Gold, 799
  Kasami, 799
  maximal-length, 797
  peak cross-correlation, 799
  preferred, 799
  (*see also* Spread spectrum signals),
Pseudocovariance
  for complex random processes, 71
PSK, 101–103, 191–195
  bit error probability, 195
  Differential (DPSK), 221
  differentially encoded, 195
  message error probability, 194
Pulse amplitude modulation (*see* PAM)
Pulsed interference, 784
  effect on error rate performance, 785–791
Punctured codes, 446, 516, 521–523
Punctured convolutional codes, 516, 521–523
  rate compatible, 523–525
Puncturing matrix, 520, 522
Pythagorian relation, 29

Q-function, 41
QAM, 103–107, 185–187, 196–200
  error probability, 196–200
QPSK, 102
  error probability, 199
  message error probability, 193
  offset (OQPSK), 124
Quadrature amplitude modulation (*see* QAM)
Quadrature component, 22
Quasi-perfect codes, 435
Quaternary PSK (QPSK), 102

R_0 (channel cutoff rate), 527, 787–791, 957–960
  For fading channels, 957–960
Raised cosine spectrum, 607
  excess bandwidth, 607
  rolloff parameter, 607
RAKE demodulator, 869–882
  for binary antipodal signals, 878
  for binary orthogonal signals, 874–877
  for DPSK signals, 878
  for noncoherent detection of orthogonal signals, 879
  generalized, 880–882
Random access,1068–1077
  ALOHA, 1069–1073
  carrier sense, 1073–1077
    with collision detection, 1073
    non persistent, 1074
    1-persistent, 1074
    p-persistent, 1074–1077
  offered channel traffic, 1070
  slotted ALOHA, 1070
  throughput, 1070
  unslotted, 1070
Random coding, 362, 375
Random processes, 66–81
  bandlimited, 74–76
  bandpass, 78–81
  cross spectral density, 67
  cyclostationary, 70
  discrete-time, 69
  Gaussian, 68
  jointly wide-sense stationary, 67
  narrowband, 79
  power, 68
  power spectral density, 67
  power spectrum, 67
  proper, 71
  sampling theorem, 74
  series expansion, 74
  white, 69
  wide-sense stationary, 67
Random variables, 40–57
  Bernoulli, 40
  binomial, 41
  characteristic function, 44
  χ^2, 45
  complex, 63
  exponential, 46
  gamma, 46
  Gaussian, 41
  iid, 45
  jointly Gaussian, 54
  lognormal, 54
  moment generating function, 44
  Nakagami, 52
  non-central χ^2, 46
  normal, 41
  Rayleigh, 48
  Ricean, 50
  uniform, 41
Random vectors,
  circular, 66
  circularly symmetric, 66
  complex, 64
  proper, 65
Rate
  bit, 97
  code, 2, 402
  signaling, 97
Rate-compatible punctured convolutional codes (RCPCC), 523–525
Rate-distortion function, 350
  Shannon's lower bound, 353
Rate-distortion theorem, 351
Rayleigh fading channel, 833, 841, 846–868
  CSI at both sides, 912
  CSI at receiver, 909, 957–960
  ergodic capacity, 907
    for MIMO channels, 985–987
  no CSI, 908
  outage capacity, 913
    for MIMO channels, 987–990
Rayleigh random variable, 48
RCC (recursive convolutional codes), 507
RCPCC (rate-compatible punctured convolutional codes), 523–525
Receiver
  correlation, 177
  MAP, 162
  matched filter, 178–182
  ML, 163, 623–625
Receiver implementation, 177
Reciprocal polynomial, 450

Recursive convolutional codes,
Recursive least squares (RLS) algorithms, 710–721
  fast RLS, 715
  RLS Kalman, 711–714
  RLS lattice, 716–721
Recursive systematic convolutional codes (RSCC), 507
Reed-Muller codes, 421
Reed-Solomon codes, 441, 446, 471–475
  burst error correction, 473
  decoding, 473
  MDS property, 472
  weight enumeration polynomial, 473
References, 1109
Regenerative repeaters, 260–261
Reliability function, 369
Reliable communication, 207, 361
Residuals, 718
Rice factor, 51
Ricean fading channel, 833,
Ricean random variable, 50–52
RS codes (*see* Reed-Solomon codes)
RSCC (*see* recursive systematic convolutional codes)

Sampling theorem, 74
Scattering function, 837
SCBC (*see* serially concatenated block codes)
Schläfli lattice, 234
Scrambling sequence, 997
Sequential decoding, 525–528
Serially concatenated block codes, 480
Set partitioning labeling, 572–573, 939
Shannon
  first theorem, 336
  lower bound on $R(D)$, 353
  second theorem, 361
  third theorem, 351
Shannon limit, 207, 554, 570
Shaping, 586
Shaping gain, 240, 586
Shortened codes, 445
Shortened cyclic codes, 452
Signal (see also Signals)
  analytic, 21
  bandpass, 21
  bandwidth, 20
  baseband, 20
  complex envelope, 22
  energy of, 25
  envelope of, 23
  fading, 8
  in-phase component, 22
  lowpass, 20
  lowpass equivalent, 22
  multipath, 8, 831
  narrowband, 18–21
  norm, 30
  parameter estimation, 290–326
  phase, 23
  quadrature components of, 22
  spectrum, 19
Signal design, 602–611, 619–623
  for band-limited channel, 602
  for channels with distortion, 619–623
  for no intersymbol interference, 604–609
  with partial response pulses, 609–611
  with raised cosine spectral pulse, 607–608
Signal constellation, 28
Signal space diversity, 928
Signal space representation, 34
Signal-to-noise ratio (SNR), 176, 192
Signaling
  based on binary codes, 113
  binary antipodal, 101
  biorthogonal, 111
  digital, 95
  multidimensional, 108
  non-return-to-zero (NRZ), 115
  non-return-to-zero, inverted (NRZI), 115
  on-off, 267
  orthogonal, 108
  simplex, 112
  with memory, 114
Signaling interval, 96
Signaling rate, 97
Signals
  antipodal, 101
  binary coded, 113
  binary orthogonal, 176–177
  biorthogonal, 111
  digitally modulated, 95
    cyclostationary, 70–71, 131
    representation of, 28, 95
    spectral characteristics, 131
  inner product, 26
  M-ary orthogonal, 108–111
  multiamplitude, 98
  multidimensional, 108–114
  multiphase, 101–103
  orthogonal, 30
  random, 66–81
    autocorrelation, 67
    bandpass stationary, 78–81
    cross correlation of, 67
    power density spectrum, 67
    properties of quadrature components, 79–81
    white noise, 69
  quadrature amplitude modulated (QAM), 103–106
  simplex, 112–113
Signature sequence, 1037
Simplex signaling, 112–113
  optimal detection, 209–210
Single-sideband (SSB) PAM, 100
Singleton bound, 440
Singular-value decomposition, 974–975, 981–982, 1087
  left singular vectors, 981, 1087
  right singular vectors, 981, 1087
  singular values, 974, 981, 1087
SISO (soft-input-soft-output) decoder, 545
Skew-Hermitian matrix, 65
Skin depth, 9
SNR, 176
  Per bit, 176
  per symbol, 192
Soft decision decoding, 424
Source 330–354
  analog, 330
  binary, 331
  discrete memoryless (DMS), 332
  discrete stationary, 337
  encoding, 339–354
    discrete memoryless, 339
    Huffman, 342–346
    Lempel-Ziv, 346–348
Source coding, 1, 339–354
Space-time codes, 1006–1021
  concatenated, 1020–1021
  differential STBC, 1014
  orthogonal STBC, 1011–1013
  quasi-orthogonal STBC, 1013
  trellis, 1016–1019
  turbo, 1020–1021
Spaced-frequency, spaced-time correlation function, 835
Spatial rate, 1007
Spectral bit rate, 226
Spectral shaping
  by precoding, 134, 611–612
Spectrum
  of CPFSK and CPM, 138–147
  of digital signals, 131–148
  of linear modulation, 133–135
  of signals with memory, 131–133, 135–147
Specular component, 841
Sphere packing, 235
Sphere packing bound, 441
Spread factor, 845
  table of, 845
Spread spectrum multiple access (SSMA), 1031
Spread spectrum signals, 763–765
  acquisition of, 816
  for code division multiple access (CDMA), 779–780, 813–814
    for MIMO systems, 992–996
  concatenated codes for, 776–778
  direct sequence, 765–768
    application of, 778–784
    coding for, 776–778
    demodulation of, 767–768
    performance of, 768–773
    with pulse interference, 784–791
  excision of narrowband interference, 791–796
  for low-probability of intercept (LPI), 778–779
  for multipath channels, 869–871, 997–1000
  frequency-hopped (FH), 802–804
    block hopping, 803
    performance of, 804–806
    with partial-band interference, 806–812
  hybrid combinations, 814–815
  interference margin, 774
  processing gain, 773–774
  synchronization of, 815–822
  time-hopped (TH), 814
  tracking of, 819–822
  uncoded DS, 775
Spread spectrum system model, 763–765
Square-law detection, 216
Square-root factorization, 715
SQPSK, 124–128
SSB, 100
Staggered QPSK (SQPSK), 124–128
Standard array, 430
State diagram, 496
Stationary random processes,
  wide-sense, 67
Stationary source, 337
Steepest-descent (gradient) algorithm, 691–701
Storage channel, 9
Subfield, 483
Sublattice, 234
Subscriber local loop, 756
Successive interference cancellation, 1047–1048
Sufficient statistics, 166
Sum-Product algorithm, 558–567
Survivor path, 244, 512
SVD (See Singular-value decomposition)
Symbol error probability, 164
Symbol rate, 97
Symbol SNR, 192
Symmetric channel capacity, 363
Synchronization
  carrier, 290–315
    effect of noise, 300–303
    for multiphase signals, 313–314
    with Costas loop, 312–315
    with decision-feedback loop, 303–308
    with phase-locked loop (PLL), 298–303
    with squaring loop, 310–312
  of spread spectrum signals, 815–822
    with tau-dither loop, 820
    with delay-locked loop, 819
  sequential search, 818
  sliding correlator, 816
  symbol, 290–291, 315, 321
Syndrome, 430, 467
  polynomial, 458
Systematic block codes, 412
Systematic convolutional codes,
Systematic cyclic codes, 453

Tail probability bounds 56–63
  Chernov bound, 58–63, 866–868
  Markov bound, 56, 57
Tanner graph 558–561
  for low density parity check codes, 569–570
TATS (tactical transmission system), 813
Telegraphy, 12
Telephone channels, 598–601
Ternary Golay code, 442
Theorem
  central limit, 63
  dimensionality, 227
  lossless source coding, 336
  Mercer, 77
  noisy channel coding, 361
  rate-distortion, 351
  Shannon's second, 361
  Shannon's third, 351
  Wiener-Khinchin, 67
Thermal noise, 3, 69
Threshold decoder, 531
Time diversity, 851
Time division multiple access (TDMA), 1030
  capacity of, 1032–1033
Timing phase, 315
Toeplitz matrix, 700
Tomlinson-Harashima precoding, 668–669
Transfer function of convolutional codes, 500
Transform domain generator matrix, 495
Transpose of a matrix, 28
Tree diagram, 496
Trellis, 116, 243, 496
Trellis-coded modulation, 571–589
  encoders for, 583
  for fading channels, 929–935
  free Euclidean distance, 577
  set partitioning, 572
  subset decoding, 578
  tables of coding gains for, 581–582
  turbo coded, 586–589
Trellis diagram, 496
Triangle inequality, 29–30
Turbo cliff region, 553
Turbo codes, 548–558
  error floor, 551
  EXIT charts, 555
  for fading channels, 1020–1021
  interleaver gain, 552
  iterative decoding, 552
  Max-Log-APP algorithm, 548
  multiplicity, 549
  turbo cliff region, 553
  waterfall region, 553
Turbo TCM, 586–589
Turbo decoding algorithm, 552
Turbo equalization, 671–673
Typical sequences, 336
Underspread fading channels, 899
Underwater acoustic channels, 9
Undetected error, 430
Unequal error protection, 523
Uniform interleaver, 480–481
Uniform random variable, 41
Union bound, 182–186
Uniquely decodable source coding, 339
Universal source coding, 347

Variable-length source coding, 339
Variance, 40
Varshamov-Gilbert bound, 443
Vector space, 28–30, 410–411
Vectors
  linearly independent, 29
  norm, 28
  orthogonal, 28
  orthonormal, 28
Viterbi algorithm, 243–246, 510–513
  path memory truncation, 246, 513
  survivor, 244–245, 512
  survivor path, 245, 512
Voltage-controlled oscillator (VCO), 298
Voronoi region
  of a lattice point, 232
Waterfall region, 553
Water-filling interpretation, 745, 902
  in time, 912
Waveform channels, 358
WEF (weight enumeration function), 415
Weight distribution, 411
Weight distribution polynomial (WEP), 415
Weight enumeration function, 415
Weight of a codeword, 411
Welch bound, 801
White processes, 69
Whitened matched filter (WMF), 627
Whitening filter, 167, 627
Wide-sense stationary process, 67
Wiener-Khinchin theorem, 67
Wireless electromagnetic channels, 5
Wireline channels, 4
Word error probability, 417
WSS (side-sense stationary), 67

Yule-Walker equations, 716

Z transform, 626
Zero-forcing equalizer, 642
Zero-forcing filter, 642